DEEP LEARNING FOR VOLUMETRIC MEDICAL IMAGE SEGMENTATION

by
Zhuotun Zhu

A dissertation submitted to Johns Hopkins University in conformity with the requirements for the degree of Doctor of Philosophy

Baltimore, Maryland
December, 2020

Abstract

Over the past few decades, medical imaging techniques, *e.g.*, computed tomography (CT), positron emission tomography (PET), have been widely used to improve the state of diagnosis, prognosis, and treatment of diseases. However, reading medical images and making diagnosis or treatment planning require well-trained medical specialists, which is labor-intensive, time-consuming, high-cost and error-prone.

With the emerging of deep learning, doctors and researchers have started to benefit from medical image analysis in various applications, *e.g.*, medical image registration, classification, detection and segmentation. Among these tasks, segmentation is the most common area of applying deep learning to medical imaging. How to improve medical diagnosis by advancing the segmentation in computer-aided diagnosis systems has become an active research topic.

In this dissertation, we will address this topic in following aspects. (i) We propose a 3D-based coarse-to-fine framework to effectively and efficiently tackle the challenges of limited amount of annotated 3D data and limited computational resources in the field of volumetric medical image segmentation. (ii) We extend the 3D coarse-to-fine to be multi-scale to early detect the small but clinically important pancreatic ductal adenocarcinoma (PDAC) tumors, and provide radiologists with interpretable abnormality locations by segmentation-for-classification. (iii) We extend the segmentation-for-classification to screen pancreatic neuroendocrine (PNETs) tumors by incorporating dual-phase information and dilated pancreatic duct that is regarded as the sign of high risk for pancreatic cancer. (iv) Going further, we investigate the mainstream methodol-

ogy in the segmentation area and then explore the novel idea of AutoML in the medical imaging field to automatically search the neural network architectures tailoring for the segmentation task, which further advances the medical image segmentation field. (v) Moving forward beyond pancreatic tumors, we are the first to address the clinically critical task of detecting, identifying and characterizing suspicious cancer metastasized lymph nodes (LNs) by proposing a 3D distance stratification strategy to simulate and simplify the high-level reasoning protocols conducted by radiation oncologists in a divide-and-conquer manner. (vi) The 3D distance stratification strategy is upgraded by our proposed multi-branch detection-by-segmentation, which further advances the finding, identifying and segmenting of metastasis-suspicious LNs.

Dissertation Readers

Dr. Alan L. Yuille (Primary Advisor)
Bloomberg Distinguished Professor
Department of Computer Science
Johns Hopkins University
IEEE Fellow

Dr. Elliot K. Fishman, M.D.
Professor
Radiology and Radiological Science
Johns Hopkins University School of Medicine

Dr. Le Lu
Executive Director
PAII Bethesda Research Lab
IEEE Fellow

Acknowledgements

First and foremost, I would like to express my sincere appreciation to my advisor Prof. Alan L. Yuille for his genuine patience, tremendous support and insightful guidance throughout my PhD career. Alan is a "Jedi Master" as an excellent mentor. There was a time back in UCLA when my English listening and speaking were poor that our research conversation could not flow. Alan did not give up the discussion but had another senior student as the translator. In the first year after our group moved to JHU, my research run far from smooth and I was stuck with using 3D convs to segment CT scans. Alan reached out to JHU professors and experts to help me sort it out since our group did not have the particular expertise at that time. Later on, this direction became my core path throughout my PhD study and will carry on. On the FELIX project, I was able to have many research discussions with Alan, either in personal or a group, where he can always pin down painpoints and propose crucial improvements. These dots and moments remind me how lucky I was to have Alan as my PhD advisor, which shapes my attitude towards research. Without Alan's advice, I could have never made current achievements.

Next, I would wholeheartedly appreciate Prof. Elliot K. Fishman for serving as my GBO and thesis committee members, who is a world-class radiologist in JHMI. My PhD career is highly dedicated to the FELIX project for early detection of pancreatic cancer using deep learning techniques, founded by Lustgarten Foundation. In this project, Elliot leads the radiologist team with one major duty to annotate the CT scans per-voxel, which is labor-intensive and time-consuming. The high-quality annotated

data makes it possible for us to design and validate our algorithms. Besides, I also want to thank Prof. Bert Vogelstein, Prof. Kenneth Kinzler, Prof. Ralph Hruban, Prof. Karen Horton, Prof. Satomi Kawamoto, Prof. Linda Chu, Prof. Seyoun Park, Dr. Eva Zinreich, Dr. Alejandra Blanco, Dr. Daniel F. Fouladi, Dr. Shahab Shayesteh, Dr. Saeed Ghandili, who are involved in this interdisciplinary project with potentially unprecedented clinical impact.

I am more than grateful to be advised by Dr. Le Lu in PAII, where I spent multiple semesters very worthily as an intern. Le has a terrific rich experience in both academia and industry in medical image analysis and oncology imaging. It was from him that I realized that a computer science researcher in medical image field should think much more beyond the computer science perspective. Only after we bear in mind that "Doctors and patients first" can we serve them well. A meaningful dedication into computer-aided diagnosis solutions should start from the clinical applications first, then brainstorm how to model, followed by what annotations we need, and finally what data we want, rather than vice versa. Without Le's fantastic mentorship, I would have never been who I am today. In PAII, I also learnt so much from many professional experts and made amazing friends here, *i.e.*, Dakai, Dazhou, Ke, Jinzheng, Adam, Ling, Shun, Jiawen, Kang, Yuankai, Bowen, Zhanghexuan, Yuhang, Yirui, Haomin, Weijian, Ashwin, and our administrative staff Yizhi. I also want to thank our collaborators outside PAII in my internships here, *i.e.*, Chun-Hung Chao, Dr. Tsung-Ying Ho, Dr. Xianghua Ye, Dr. Chen-Kan Tseng, and Dr. Tzu-Chen Yen.

I am also glad to work with Dr. Daguang Xu and Dr. Dong Yang in NVIDIA, and Dr. Waoyuan Wang in MSR. They are great mentors and collaborators.

Besides, I would like to thank Prof. Greg Hager, Prof. Wei Shen, Prof. Alex Szalay, Prof. Vishal Patel and Prof. Ralph Etienne-Cummings for taking their valuable time in participating my PhD GBO exam and providing valuable suggestions and comments to my coursework and research study. What's more, I want to thank my master thesis

committee Prof. Yingnian Wu and Prof. Hongjing Lu back in UCLA, and Prof. Xiang Bai and Prof. Xinggang Wang who led me to this magical research world back in HUST when I was an undergrad student.

Last but not the least, during my PhD study, I am fortunate to meet and/or work with so many nice and professional postdocs, fellow PhD students and visiting students in AlanLab, *i.e.*, Lingxi, Yan, Yongyi, Ehsan, Adam, Weichao, Chenxi, Siyuan, Zhishuai, Cihang, Yuyin, Chenxu, Huiyu, Qing, Qi, Fengze, Yingda, Yi, Hongru, Qihang, Jieru, Yingwei, Yixiao, Zhuowan, Zihao, Chenglin, Yutong, Angtian, Chen, Jieneng, Peng, Song, Feng, Chang, Shuhao, and also other people back in UCLA, *i.e.*, Jun, Vittal, Liang-Chieh, Xianjie, Zhou, Alex, Fangting, Junhua, Peng, Jianyu, Xiaochen and John. Not only do we discuss research with each other, but also we play and eat together. Those happy memories won't be forgotten.

To my beloved parents and family for their unconditional and unreserved support.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Over the past few decades, medical imaging techniques, *e.g.*, magnetic resonance imaging (MRI), computed tomography (CT), positron emission tomography (PET), ultrasound, and X-ray, have been widely used to improve the state of medical diagnosis, prognosis and treatment. However, reading medical images and making diagnosis or treatment planning require well-trained medical specialists. Given the wide variation in medical imaging and potential fatigue of human experts, these tasks are labor-intensive, time-consuming, high-cost and error-prone.

Deep learning has a game-changing potential to improve the state of preventative and precision medicine within medical image computing. In medical imaging, preventative medicine refers to early detection of disease findings, *e.g.*, colonic polyps [1], lung nodules or liver/bone lesions [2], with the goal of timely patient intervention and management [3]. It is traditionally done by medical experts through manual examination on non-invasive imaging modalities, but more recently computer-aided diagnosis systems are coming into prominence. Within imaging, the precision medicine stands for quantitatively and precisely computing imaging biomarkers, *e.g.*, volumetric tumor measurements for tracking and beyond, to improve clinical decision making and patient outcomes [3]. Current radiological practices are still largely qualitative, but automated quantitative assessment shows a significantly better surrogate endpoint than human assessment for predicting overall survival [4]. Therefore, advanced au-

tomatic computer-aided diagnosis solutions could ultimately serve as a blueprint to improve the overall survival.

Extracting effective features for medical image analysis problems is notoriously hard for hand-crafted designs whereas deep learning is the state-of-the-art technique for automatically learning features in a data-driven perspective and achieves this goal. With the emerging of deep learning, doctors and researchers have started to benefit from medical image analysis in various applications, *e.g.*, medical image registration [5][6], classification [7][8], detection [2][9], segmentation [10][11] and other tasks [12][13]. Among these tasks, segmentation is the most common area of applying deep learning to medical imaging [14].

In medical image, segmentation is the process of partitioning an image into different segments, with these segments corresponding to different tissues, organs, lesions, pathologies, or other biologically relevant structures. Medical image segmentation is made difficult by low contrast, high variations, inevitable noise, and other imaging ambiguities, *e.g.*, missing edges, indistinguishable textures, region of interest (ROI) messed into clustered backgrounds. Although challenging, it is a prerequisite step for many clinical applications, such as diabetes inspection [15], cancer diagnosis [16], and surgical planning [17]. Therefore, it is well worth exploring automatic segmentation to advance the computer-aided diagnosis systems.

In this dissertation, we advance the volumetric medical image segmentation via state-of-the-art deep learning techniques to improve the state of medical diagnosis.

In one aspect, we explore how to segment organs and/or tumors more accurate, as presented in chapters from Chapter 3 to Chapter 6. Take, for example, the pancreas segmentation with its extension to early detection and segmentation of pancreatic tumors as shown in Fig. 1.1, we visualize several contrast-enhanced CT cases from online open dataset (NIH normal pancreas [18] and MSD abnormal pancreas [19]),

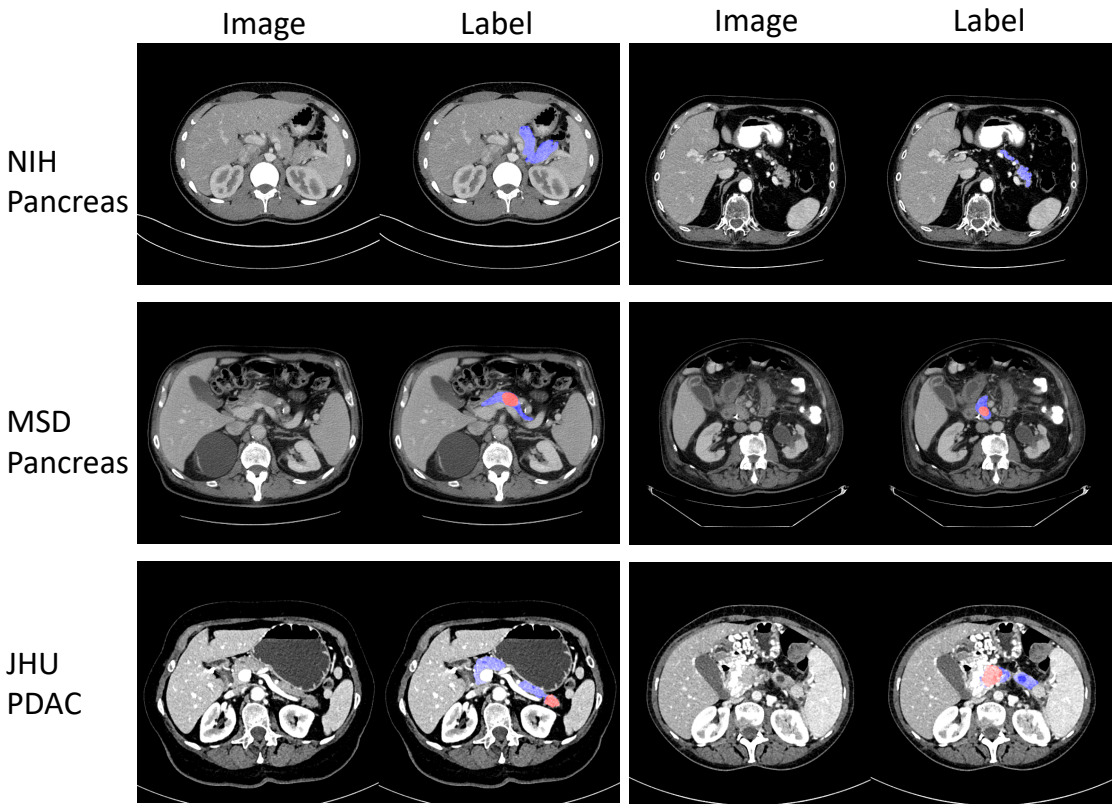

Figure 1.1. Typical examples from NIH Pancreas [18] in the 1st row, MSD Pancreas Tumors [19] in the 2nd row, and JHU PDAC pancreas [20] in the 3rd row. Two slices of different cases are randomly chosen from each dataset. Normal Pancreas regions are masked as blue and abnormal pancreas regions are masked as red. Best viewed in color.

and our in-house abnormal pancreas dataset (JHU PDAC [20]). 95% of all pancreatic cancer cases are pancreatic adenocarcinoma (PDAC) [21], which is the most common type of pancreatic cancers. In the early stage of pancreatic cancers, it often has few symptoms and is very difficult to discover. By the time of diagnosis, the cancer has often spread to other parts of the body, leading to a very poor prognosis (a five-year survival rate of 5% [22]). But, for cases diagnosed early, the survival rate rises to about 20% [23]. Hence, it is very important to study the possibility of segmenting and detecting both normal pancreas and abnormal pancreas in common examinations, *e.g.*, the abdominal CT scan. The main challenges of pancreas segmentation and the early detection of pancreatic cancers lie in several aspects: 1) the small size of targets with

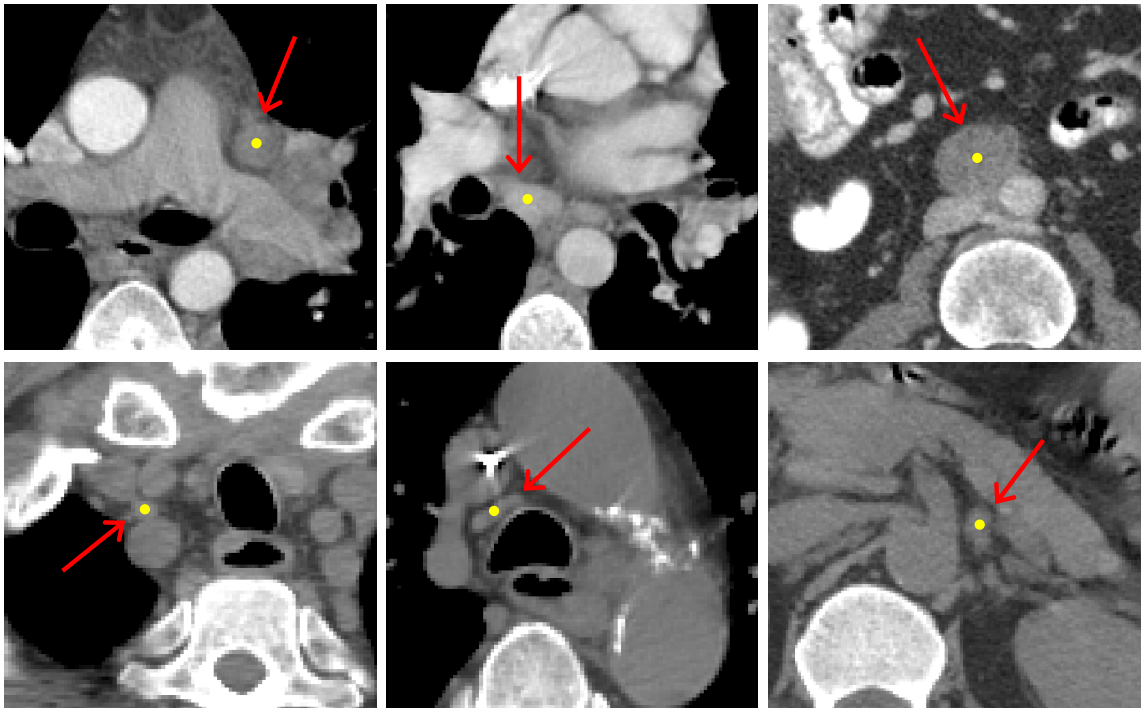

Figure 1.2. Typical examples with varying size and appearance at scatteredly distributed locations from the in-house GTV_{LN} dataset [24]. The red arrow points out the location of GTV_{LN} with a yellow dot indicates the position. Best viewed in color.

respect to the whole volume; 2) the large variations in location, shape and appearance across different cases; 3) the abnormalities, *i.e.*, the pancreas tumors, can change the texture of surrounding tissues a lot; 4) the anisotropic property along z-axis, which make the automatic segmentation even harder.

In another aspect, we are the first to computationally address the clinically critical task of segmenting, identifying and characterizing suspicious cancer metastasized lymph nodes (termed as oncology-significant lymph nodes, abbreviated as OSLNs, or lymph node gross tumor volume, abbreviated as GTV_{LN}) in Chapter 7 and Chapter 8. It is an extremely challenging and time-consuming task to identify GTV_{LN}, even for experienced radiation oncologists. High-level sophisticated clinical reasoning guidelines are needed, leading to the risk of uncertainty and subjectivity with high inter-observer variabilities [25]. Refer to Fig. 1.2 as an illustration of GTV_{LN}, the challenges of

this task are lying in following aspects. (1) Finding GTV_{LN} is often performed using radiotherapy CT (RTCT) that (unlike diagnostic CT) is not contrast-enhanced. Hence the metastasis signs for identifying GTV_{LN} are subtle. (2) GTV_{LN} itself has poor contrast. Because of the shape and appearance ambiguity, it can be easily confused with vessels or muscles. (3) The size and shape of GTV_{LN} vary considerably with large amounts of smaller ones that are harder to detect. As far as we know, no prior work, as of yet, has computationally studied the GTV_{LN} segmentation and detection in non-contrast RTCT scans.

1.1 Challenges and Our Contributions

1.1.1 Volumetric Pancreas Segmentation

With the rapid development of deep neural networks and their success in many computer vision tasks [26]–[28], many deep learning based methods have been proposed for pancreas segmentation and achieved considerable progress [18], [29], [30]. However, these methods are based on 2D fully convolutional networks (FCNs) [31], which perform segmentation slice by slice whereas CT volumes are indeed 3D data. Using 3D deep networks for organ segmentation [10], [11], [32]–[34], is a recent trend but not yet applied to the pancreas. An obstacle is that training 3D deep networks suffers from the "out of memory" problem. 2D FCNs can accept a whole 2D slice as input, but 3D FCNs cannot be fed a whole 3D volume due to the limited GPU memory size. A common solution is to train 3D FCNs from small sub-volumes and test them in a sliding-window manner, *i.e.*, performing 3D segmentation on densely and uniformly sampled sub-volumes one by one. Usually, these neighboring sampled sub-volumes overlap with each other to improve the robustness of the final 3D results. It is worth noting that the overlap size is a trade-off between the segmentation accuracy and the time cost. Setting a larger/smaller overlap size generally leads to a better/worse segmentation accuracy but takes more/less time during testing. To address this

challenge, we propose a **3D coarse-to-fine** framework to first obtain the rough location of the target pancreas from the whole CT volume by a 3D FCN trained on whole CT volumes, and then refine the segmentation by another 3D FCN trained on the pancreas regions only.

To our best knowledge, we are one of the first studies to segment the challenging ***normal*** and ***abnormal*** pancreases using 3D networks which leverage the rich spatial information. The effectiveness and efficiency of the proposed 3D coarse-to-fine framework are demonstrated on two pancreas segmentation datasets where we achieve the state-of-the-art with relative low time cost. It is worth mentioning that, although our focus here is pancreas segmentation, our framework is generic and can be directly applied to segmenting other medical organs.

1.1.2 Pancreatic Tumors Segmentation for Classification

Next, we extend the aforementioned **3D coarse-to-fine** framework to segment PDAC or PNETs from a mixture of normal and abnormal CT scans. At a first glance, we need to classify each CT scan to be normal or abnormal. This is not a simple classification task since radiologists also want to know the location of pancreatic tumors, we suggest a solution named **segmentation-for-classification**, which trains segmentation models and uses their outputs for classification. The difficulty mainly lies in the tiny size, irregular shape and low contrast around the boundary of pancreatic tumors. In segmenting PDAC, to deal with tumors of various sizes, we adopt a segmentation network with multiple input scales, *i.e.*, 64^3, 32^3 and 16^3 volumes. But, voting that small input regions lead to a high false alarm rate, we adopt a **coarse-to-fine** testing strategy, which uses the 64^3 network for a coarse scan, and then the 32^3&16^3 networks inside the bounding box to detect small tumors that are possibly ignored in the previous stage. A non-parameterized post-processing algorithm is designed to remove outliers. We name this framework as **multi-scale coarse-to-fine**

to do **segmentation-for-classification**. In segmenting PNETs, on the one hand, to deal with the observation that some PNETs are only visible to radiologists in either phase (venous or arterial phase), the dual-phase information is straightforwardly combined to reduce the missing detection of PNETs. On the other hand, radiologists are very sensitive to the dilated pancreatic duct when reading CT scans. There are often occasions the pancreatic duct is visible to be dilated although the PNETs regions are barely visible from CT appearance and texture. So in order to incorporate this knowledge to our framework, we annotate dilated pancreatic duct as well and segment them in the meantime, which is regarded as the sign of high risk for pancreatic cancer.

Our contributions in segmenting PDACs or PNETs are: 1) we voxelwisely annotate a PDAC and PNET dataset, which are currently the largest pancreatic tumors dataset to the best of our knowledge; 2) we adopt a **segmentation-for-classification** framework to conduct an **interpretable** abnormality detection, which provides radiologists with suspicious regions for further diagnosis; 3) our framework achieves a high sensitivity and specificity in terms of binary abnormal classification, which shows a promising direction to make a potential significant clinical impact.

1.1.3 Neural Architecture Search for Medical Image Segmentation

Going one step further, we investigate the mainstream methodology in the segmentation area and then explore the novel idea of NAS/AutoML in medical imaging field to develop the state-of-the-art segmentation network. The well-known fully convolutional neural networks (FCNs) [31], *e.g.*, 2D and 3D FCNs, deliver powerful representation ability and good invariant properties. The 2D FCNs based methods [18], [29], [30], [35], [36] perform the segmentation slice-by-slice from different views, then fuse 2D segmentation output to obtain a 3D result, which is a remedy against the ignorance of the rich spatial information. To make full use of the 3D context, 3D FCNs based

methods [10], [11], [37], [38] directly perform the volumetric prediction. However, the demanding computation and high GPU consumption of 3D convolutions limit the depth of neural networks and input volume size, which impedes the massive application of 3D convolutions. Recently, the Pseudo-3D (P3D) [39] was introduced to replace 3D convolution $k\times k\times k$ with two convolutions, *i.e.*, $k\times k\times 1$ followed by $1\times 1\times k$, which can reduce the number of parameters and show good learning ability in [40], [41] on anisotropic medical images. However, all the aforementioned existing works choose the network structure empirically, which often impose explicit constraints, *i.e.*, either 2D, 3D or P3D convolutions only, or 2D and 3D convolutions are separate from each other. These hand-designed segmentation networks with architecture constraints might not be the optimal solution considering either the ignorance of the rich spatial information for 2D or suffering from the demanding computations for 3D. Drawing inspiration from recent success of Neural Architecture Search (NAS), we take one step further to let the segmentation network **automatically** choose between 2D, 3D, or P3D convolutions at each layer by formulating the structure learning as **differentiable neural architecture search** [42], [43].

To the best of our knowledge, we are one of the first to explore the idea of NAS/AutoML in medical imaging field. Previous work [44] used reinforcement learning and the search restricts to 2D based methods, whereas we use differentiable NAS and search between 2D, 3D and P3D, which is more effective and efficient. Without pretraining, our searched architecture, named V-NAS, outperforms other state-of-the-arts on segmentation of normal pancreas, the abnormal lung tumors and pancreatic tumors. In addition, the searched architecture on one dataset can be well generalized to others, which shows the robustness and potential clinical use of our approach.

1.1.4 Metastasis-Suspicious Lymph Node Detection by Segmentation

Measuring lymph nodes (LNs) size and assessing its status are important clinical tasks, usually used to monitor cancer diagnosis and treatment responses and to identify treatment areas for radiotherapy. According to the Revised RECIST guideline [45], [46], only enlarged LNs with a short axis more than 10-15 mm in computed tomography (CT) images should be considered as abnormal. Such enlarged LNs have been the only focus, so far, of LN segmentation and detection works [7], [47]–[53]. However, in cancer treatment, besides the primary tumor, all metastasis-suspicious LNs are required to be treated. This includes the enlarged LNs, as well as smaller ones that are associated with a high positron emission tomography (PET) signal or any metastasis signs in CT. This larger category is regarded as lymph node gross tumor volume, abbreviated as GTV_{LN}. Identifying the GTV_{LN} and assessing their spatial relationship and causality with the primary tumor is a key requirement for a desirable cancer treatment outcome [54].

Identifying GTV_{LN} can be a daunting and time-consuming task, even for experienced radiation oncologists. It requires using high-level sophisticated reasoning protocols and faces strong uncertainty and subjectivity with high inter-observer variability [25]. To the best of our knowledge, this problem has not been previously tackled in a fully automatized way. Our task on GTV_{LN} detection is more challenging for the following reasons: (1) Finding GTV_{LN} is often performed using radiotherapy CT (RTCT), which, unlike diagnostic CT, is not contrast-enhanced. (2) GTV_{LN} exhibit low contrast with surrounding tissues and can be easily confused with other anatomical structures, *e.g.*, vessels and muscles, due to shape and appearance ambiguity. (3) The size and shape of GTV_{LN} can vary considerably, and GTV_{LN} are often scatteredly distributed at small size in a large spatial range of anatomy locations. There is an observation that GTV_{LN} has higher frequencies at smaller sizes than enlarged LNs,

challenging its detection. While, many previous works proposed automatic detection systems for enlarged LNs in contrast-enhanced CT [2], [7], [47], [48], [51], [53], [55], no work, as of yet, has focused on OSLN detection on non-contrast RTCT. Given the considerable differences between enlarged LNs and GTV_{LN}, further innovation is required for robust and clinically useful GTV_{LN} detection.

Our contributions can be summarized as follows: 1) To the best of our knowledge, we are the first to address the clinically critical task of detecting, identifying and characterizing GTV_{LN}. 2) We propose a novel 3D distance stratification strategy to divide and conquer the complex distribution of GTV_{LN} into tumor-proximal and tumor-distal sub-categories, to be solved separately, which emulates physician's decision process. 3) Besides RTCT, we incorporate the PET imaging modality and 3D tumor distance maps into our detection-by-segmentation network. 4) We propose a novel GLNet to incorporate high-level ontology-derived semantic attributes of GTV_{LN} with localized features computed from RTCT/PET. 5) We collect and evaluate on the largest dataset to date on chest and abdominal radiotherapy. Our dataset comprises of 651 voxelwise-labeled GTV_{LN} (by board-certified radiation oncologists) of 141 esophageal cancer patients. Our system improves the detection recall compared against the previous state-of-the-art CT-based detection method.

1.2 Dissertation Statement

Medical image segmentation is an important step in computer-aided diagnosis pipelines. By designing advanced network backbone/architectures and/or migrating rich knowledge from routine works of medical experts into automatic computer-aided diagnosis solutions can improve medical diagnosis.

1.3 Overview

The overview of this dissertation is illustrated as the following.

In Chapter 1 (this chapter), we introduce the topic of medical medical image segmentation, including definition, justification and application. We discuss the underlying challenges and our contributions to this dissertation topic.

In Chapter 2, we summarize previous works on the medical image analysis, especially the prevailing segmentation task for different targets and applications.

In Chapter 3, we discuss a 3D coarse-to-fine framework to deal with the limited amount of computational resources for the volumetric medical image segmentation.

In Chapter 4, we propose a multi-scale segmentation for classification to detect the lethal pancreatic ductal adenocarcinoma tumors from CT images.

In Chapter 5, we extend the segmentation for classification framework to detect the pancreatic neuroendocrine tumors from dual-phase CT images.

In Chapter 6, we propose to automatically design the network architecture tailoring to the volumetric medical image segmentation problem.

In Chapter 7, we make an first attempt to find and identify small but critically important objects, *i.e.*, suspicious cancer metastasized lymph nodes, from 3D multi-modality images via a divide-and-conquer decision stratification approach.

In Chapter 8, we build a multi-branch detection-by-segmentation network via a distance based stratification to improve the finding, identifying and segmenting of the suspicious cancer metastasized lymph nodes from 3D multi-modality imaging.

In Chapter 9, we summarize and conclude this dissertation.

1.4 Relevant Publications

The following publications or pre-prints compose the ideas in this dissertation. The "*" indicates equal contribution.

1. **Z. Zhu**, Y. Xia, W. Shen, E. K. Fishman, and A. L. Yuille, "A 3d coarse-to-fine framework for volumetric medical image segmentation," in *3DV*, 2018.

2. **Z. Zhu***, Y. Xia*, L. Xie, E. K. Fishman, and A. L. Yuille, "Multi-scale coarse-to-fine segmentation for screening pancreatic ductal adenocarcinoma," in *MICCAI*, 2019.

3. **Z. Zhu**, Y. Lu, W. Shen, E. K. Fishman, and A. L. Yuille, "Segmentation for classification of screening pancreatic neuroendocrine tumors," *arXiv:2004.02021*, 2020.

4. **Z. Zhu**, C. Liu, D. Yang, A. L. Yuille, and D. Xu, "V-NAS: neural architecture search for volumetric medical image segmentation," in *3DV*, 2019.

5. **Z. Zhu**, K. Yan*, D. Jin*, J. Cai, T.Y. Ho, A. Harrison, D. Guo, C.H. Chao, X. Ye, J. Xiao, A. L. Yuille, and L. Lu, "Detecting scatteredly-distributed, small, and critically important objects in 3d oncology imaging via decision stratification," *arXiv:2005.13705,* 2020.

6. **Z. Zhu**, D. Jin, K. Yan, T.Y. Ho, X. Ye, D. Guo, C.H. Chao, J. Xiao, A. L. Yuille, and L. Lu, "Lymph node gross tumor volume detection and segmentation via distance-based gating using 3d ct/pet imaging in radiotherapy," in *MICCAI*, 2020.

The following publications contribute to, or provide contexts and backgrounds for the ideas in this dissertation. The "*" indicates equal contribution.

1. **Z. Zhu**, L. Xie, and A. L. Yuille, "Object recognition with and without objects," in *IJCAI*, 2017.

2. Y. Xia, L. Xie, F. Liu, **Z. Zhu**, E. K. Fishman, and A. L. Yuille, "Bridging the gap between 2d and 3d organ segmentation with volumetric fusion net," in *MICCAI*, 2018.

3. Y. Li*, **Z. Zhu***, Y. Zhou, Y. Xia, W. Shen, E. K. Fishman, and A. L. Yuille, "Volumetric medical image segmentation: a 3d deep coarse-to-fine framework and its adversarial examples," *Deep Learning and Convolutional Neural Networks for Medical Imaging and Clinical Informatics,* pp. 69-91, 2019.

4. D. Guo, D. Jin, **Z. Zhu**, T.Y. Ho, A. Harrison, C.H. Chao, J. Xiao, and L. Lu, "Organ at risk segmentation for head and neck cancer using stratified learning and neural architecture search," in *CVPR*, 2020.

5. C.H. Chao, **Z. Zhu***, D. Guo*, K. Yan*, T.Y. Ho, J. Cai, A. Harrison, X. Ye, J. Xiao, A. L. Yuille, M. Sun, L. Lu, and D. Jin, "Lymph node gross tumor volume detection in oncology imaging via relationship learning using graph neural network," in *MICCAI*, 2020.

Chapter 2

Related Work

In this chapter, we discuss related works in literature in the scope of general medical image segmentation, neural architecture search, lymph node detection and segmentation.

2.1 General Medical Image Segmentation

In the medical image segmentation field, starting from handcrafted features, there are methods proposed to use region growing [56], intensity thresholding [57], and deformable models [58], which however often suffer from the limited feature representation ability and are less invariant to the large organ/lesion variations. The medical image analysis community is facing a revolution brought by the fast development of deep networks [59], [60]. Deep Convolutional Neural Networks (CNNs) based methods have dominated the research area of medical image segmentation in the last few years. Generally speaking, CNN-based methods for medical image segmentation can be divided into three major categories: 2D CNNs based, 3D CNNs based and 2D and 3D CNNs fusion.

2.1.1 2D CNNs for Segmentation

2D CNNs based methods [18], [29], [36] performed volumetric segmentation slice by slice from different views, and then fused the 2D segmentation results to obtain a 3D Volumetric Segmentation result. In the early stage, the 2D segmentation based models were trained from image patches and tested in a patch by patch manner [18], which is time consuming. Since the introduction of fully convolution networks (FCNs) [31], almost all the 2D segmentation methods are built upon 2D FCNs to perform holistic slice segmentation during both training and testing. Roth *et al* [29] performed Pancreas segmentation by a holistic learning approach, which first segment pancreas regions by holistically-nested networks [27] and then refine them by the boundary maps obtained by robust spatial aggregation using random forest. The U-Net [36] is one of the most popular FCN architectures for medical image segmentation, which is a encoder-decoder network with additional short connection between encoder and decoder paths. Based on the fact that a pancreas only takes up a small fraction of the whole scan, Zhou *et al.* [30] proposed to find the rough pancreas region and then learn a FCN based fixed-point model to refine the pancreas region iteratively. Their method is also based on a coarse-to-fine framework, but it only considered coarse-to-fine RoIs. Besides coarse-to-fine RoIs, our coarse-to-fine method also takes coarse-to-fine overlap sizes into account, which is designed specifically for efficient 3D inference.

2.1.2 3D CNNs for Segmentation

Although 2D CNNs based methods achieved considerable progresses, they are not optimal for medical image segmentation, as they cannot make full use of the 3D context encoded in volumetric data. Several 3D CNNs based segmentation methods have been proposed. The 3D U-Net [10] extended the previous 2D U-Net architecture [36] by replacing all 2D operations with their 3D counterparts. Based on the architecture of the 3D U-Net, the V-Net [11] introduced residual structures [61] (short term skip

connection) into each stage of the network. Chen *et al* [33] proposed a deep voxel-wise residual network for 3D brain segmentation. Both I2I-3D [62] and 3D-DSN [37] included auxiliary supervision via side outputs into their 3D deep networks. Despite the success of 3D CNNs as a technique for segmenting the target organs, such as prostate [11] and kidney [10], very few techniques have been developed for leveraging 3D spatial information on the challenging pancreas segmentation. Gibson *et al.* [63] proposed the DenseVNet which is however constrained to have shallow encoders due to the computationally demanding dense connections.

2.1.3 2D and 3D CNNs Fusion for Segmentation

A few recent works have been proposed to combine 2D and 3D FCNs as a compromise to leverage the advantages of both sides. VFN [64] adopted a 3D FCN by feeding the segmentation predictions of 2D FCNs as input together with 3D images. H-DenseUNet [65] hybridized a 2D DenseUNet for extracting intra-slice features and a 3D counterpart for aggregating inter-slice contexts. Most recently, nnUNet [66] proposed to ensemble 2D U-Net, 3D U-Net, and cascaded 3D U-Net in order to deal with the data diversity. However, 2D CNNs and 3D CNNs are not optimized at the same time in [64]–[66].

2.2 Neural Architecture Search

Neural Architecture Search (NAS) is the process of automatically discovering better neural architectures than human designs. We summarize the progress in along two dimensions: search algorithm and dataset/task.

Many NAS algorithms belong to either reinforcement learning or evolutionary algorithm. In the reinforcement learning formulation [67], the actions generated by an agent define the network architecture, and the reward is the accuracy on the validation set. In the evolutionary formulation [68], architectures are mutated to

produce better offsprings, again measured by validation accuracy. Although these algorithms are general, they are usually computationally costly. To address this problem, [69] progressively expand the search space in order to achieve better sample efficiency. Differentiable NAS approaches [42], [43], [70] utilize sharing among candidate architectures, and are arguably the most efficient family of algorithms to date.

At the same time, we also notice that the earlier papers [71]–[73] focused solely on MNIST or CIFAR10 dataset. Later, [67]–[69] searched for "transferable architectures" from the smaller CIFAR10 to the much larger ImageNet dataset. More recently, [74], [75] demonstrated the possibility to directly search for architectures on the ImageNet dataset. Finally, [42] extended NAS beyond image classification to semantic segmentation.

2.3 Lymph Node Detection and Segmentation

2.3.1 Generic Lesion Detection

There are two popular approaches for generic lesion detection: end-to-end [76]–[79] and two-stage methods [80]–[83]. End-to-end methods have been extensively applied to the universal lesion detection task in the largest general lesion dataset currently available, *i.e.*, DeepLesion [2], and achieved encouraging performance. Notably, a multi-task universal lesion analysis network (MULAN) [78] so far achieves the best detection accuracy using a 3D feature fusion strategy and Mask R-CNN [84] architecture.

In contrast, two-stage methods explicitly divide the detection task into candidate generation and FP reduction steps. The first step generates the initial candidates at a high recall and FP rate and the second step focuses on reducing the FP rate (especially the difficult ones) while maintaining a sufficient high recall. It decouples the task into easier sub-tasks and allows for the optimal design of each sub-task, which has shown to be more effective in problems like lung nodule [80], [83] and brain

lacune [81] detection as compared to the one-stage method. We adopt the two-stage strategy for the OSLN detection to effectively incorporate different features, *i.e.*, PET imaging, tumor distance map and high-semantic lesion attributes, into each stage. We demonstrate the necessity of our strategy by comparing with the state-of-the-art (SOTA) universal lesion detector MULAN [78] in the experiment.

2.3.2 Lymph Node Detection and Segmentation

All previous works focus only on enlarged LN detection and segmentation in contrast-enhanced CT. Conventional statistical learning approaches [48]–[50], [55] employ hand-crafted image features, such as shape, spatial priors, Haar filters, and volumetric directional difference filters, to capture LN appearance and location. More recent deep learning methods achieve better performance. [47], [51], [52] applies the FCN or Mask R-CNN to directly segment LNs. In contrast, [7], [53] proposed a 2.5D patch-based convolutional neural network (CNN) with random view aggregation to classify LNs given all LN candidates already detected, and achieves state-of-the-art (SOTA) classification accuracy for enlarged LNs. In Chapter 7, we demonstrate the effectiveness of the local and global modules in our GLNet compared with the 2.5D classification method [7].

Chapter 3

A 3D Coarse-to-Fine Framework for Volumetric Medical Image Segmentation

In this chapter, we adopt 3D Convolutional Neural Networks to segment volumetric medical images. Although deep neural networks have been proven to be very effective on many 2D vision tasks, it is still challenging to apply them to 3D tasks due to the limited amount of annotated 3D data and limited computational resources. We propose a novel 3D-based coarse-to-fine framework to ***effectively*** and ***efficiently*** tackle these challenges. The proposed 3D-based framework outperforms the 2D counterpart to a large margin since it can leverage the rich spatial information along all three axes. We conduct experiments on two datasets which include healthy and pathological pancreases respectively, and achieve the current state-of-the-art in terms of Dice-Sørensen Coefficient (DSC). On the NIH pancreas segmentation dataset, we outperform the previous best by an average of over 2%, and the worst case is improved by 7% to reach almost 70%, which indicates the reliability of our framework in clinical applications.

3.1 Introduction

Driven by the huge demands for computer-aided diagnosis systems, automatic organ segmentation from medical images, such as computed tomography (CT) and magnetic resonance imaging (MRI), has become an active research topic in both the medical image processing and computer vision communities. It is a prerequisite step for many clinical applications, such as diabetes inspection, organic cancer diagnosis, and surgical planning. Therefore, it is well worth exploring automatic segmentation systems to accelerate the computer-aided diagnosis in medical image analysis.

In this chapter, we focus on pancreas segmentation from CT scans, one of the most challenging organ segmentation problems [30][18]. As shown in Fig. 3.1, the main difficulties stem from three aspects: 1) the small size of the pancreas in the whole abdominal CT volume; 2) the large variations in texture, location, shape and size of the pancreas; 3) the abnormalities, like pancreatic cysts, can alter the appearance of pancreases a lot.

Following the rapid development of deep neural networks [59][60] and their successes in many computer vision tasks, such as semantic segmentation [31][26], edge detection [85][27][86] and 3D shape retrieval [28][87], many deep learning based methods have been proposed for pancreas segmentation and achieved considerable progress [30][18][29]. However, these methods are based on 2D fully convolutional networks (FCNs) [31], which perform segmentation slice by slice while CT volumes are indeed 3D data. Although these 2D methods use strategies to fuse the output from different 2D views to obtain 3D segmentation results, they inevitably lose some 3D context, which is important for capturing the discriminative features of the pancreas with respect to background regions.

Using 3D deep networks for organ segmentation is a recent trend but not yet applied to the pancreas. An obstacle is that training 3D deep networks suffers from

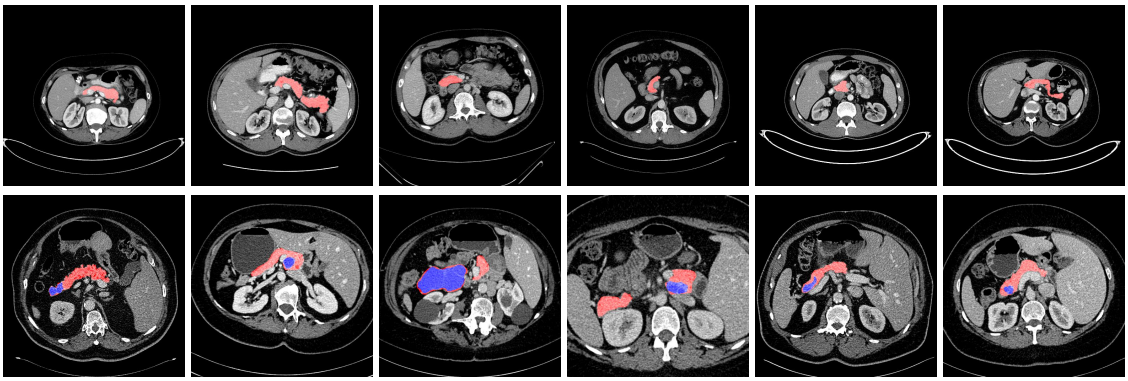

Figure 3.1. An illustration of normal pancreases on NIH dataset [18] and abnormal cystic pancreases on JHMI dataset [88] shown in the first and second row respectively. Normal pancreas regions are masked as red and abnormal pancreas regions are marked as blue. The pancreas usually occupies a small region in a whole CT scan. Best viewed in color.

the "out of memory" problem. 2D FCNs can accept a whole 2D slice as input, but 3D FCNs cannot be fed a whole 3D volume due to the limited GPU memory size. A common solution is to train 3D FCNs from small sub-volumes and test them in a sliding-window manner [11][32][10][33][34], *i.e.*, performing 3D segmentation on densely and uniformly sampled sub-volumes one by one. Usually, these neighboring sampled sub-volumes overlap with each other to improve the robustness of the final 3D results. It is worth noting that the overlap size is a trade-off between the segmentation accuracy and the time cost. Setting a larger/smaller overlap size generally leads to a better/worse segmentation accuracy but takes more/less time during testing.

To address these issues, we propose a concise and effective framework based on 3D deep networks for pancreas segmentation, which can simultaneously achieve high segmentation accuracy and low time cost. Our framework is formulated in a coarse-to-fine manner. In the training stage, we first train a 3D FCN from the sub-volumes sampled from an entire CT volume. We call this ***ResDSN Coarse*** model, which aims to obtain the rough location of the target pancreas from the whole CT volume by making full use of the overall 3D context. Then, we train another 3D FCN from the sub-volumes sampled only from the ground truth bounding boxes of the target

pancreas. We call this the ***ResDSN Fine*** model, which can refine the segmentation based on the coarse result. In the testing stage, we first apply the coarse model in the sliding-window manner to a whole CT volume to extract the most probable location of the pancreas. Since we only need a rough location for the target pancreas in this step, the overlap size is set to a small value. Afterwards, we apply the fine model in the sliding-window manner to the coarse pancreas region, but by setting a larger overlap size. Thus, we can efficiently obtain a fine segmentation result and we call the coarse-to-fine framework by ***ResDSN C2F***.

Note that, the meaning of "coarse-to-fine" in our framework is twofold. First, it means the input region of interests (RoIs) for ***ResDSN Coarse*** model and ***ResDSN Fine*** model are different, i.e., a whole CT volume for the former one and a rough bounding box of the target pancreas for the latter one. We refer to this as coarse-to-fine RoIs, which is designed to achieve better segmentation performance. The coarse step removes a large amount of the unrelated background region, then with a relatively smaller region to be sampled as input, the fine step can much more easily learn cues which distinguish the pancreas from the local background, i.e., exploit local context which makes it easier to obtain a more accurate segmentation result. Second, it means the overlap sizes used for ***ResDSN Coarse*** model and ***ResDSN Fine*** model during inference are different, i.e., small and large overlap sizes for them, respectively. We refer to this as coarse-to-fine overlap sizes, which is designed for efficient 3D inference.

To our best knowledge, we are one of the first studies to segment the challenging ***normal*** and ***abnormal*** pancreases using 3D networks which leverage the rich spatial information. The effectiveness and efficiency of the proposed 3D coarse-to-fine framework are demonstrated on two pancreas segmentation datasets where we achieve the state-of-the-art with relative low time cost. It is worth mentioning that, although our focus is pancreas segmentation, our framework is generic and can be directly applied to segmenting other medical organs.

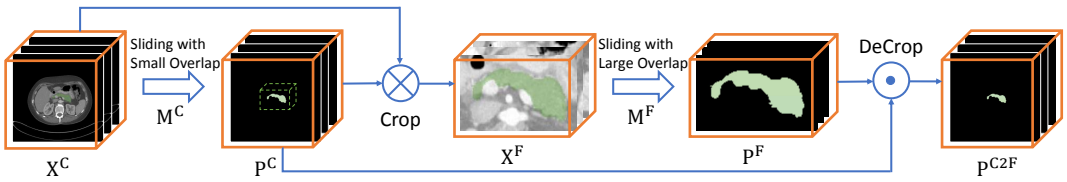

Figure 3.2. Flowchart of the proposed 3D coarse-to-fine segmentation system in the testing phase. We first apply "ResDSN Coarse" with a small overlapped sliding window to obtain a rough pancreas region and then use the "ResDSN Fine" model to refine the results with a large overlapped sliding window. Best viewed in color.

3.2 Related Work

The medical image analysis community is facing a revolution brought by the fast development of deep networks [59][60]. Deep Convolutional Neural Networks (CNNs) based methods have dominated the research area of volumetric medical image segmentation in the last few years. Generally speaking, CNN-based methods for volumetric medical image segmentation can be divided into two major categories: 2D CNNs based and 3D CNNs based.

3.2.1 2D CNNs for Volumetric Segmentation

2D CNNs based methods [18][29][89][90][36] performed volumetric segmentation slice by slice from different views, and then fused the 2D segmentation results to obtain a 3D Volumetric Segmentation result. In the early stage, the 2D segmentation based models were trained from image patches and tested in a patch by patch manner [18], which is time consuming. Since the introduction of fully convolution networks (FCNs) [31], almost all the 2D segmentation methods are built upon 2D FCNs to perform holistic slice segmentation during both training and testing. Havaei *et al* [89] proposed a two-pathway FCN architecture, which exploited both local features as well as more global contextual features simultaneously by the two pathways. Roth *et al* [29] performed Pancreas segmentation by a holistic learning approach, which first segment pancreas regions by holistically-nested networks [27] and then refine them by the boundary

maps obtained by robust spatial aggregation using random forest. The U-Net [36] is one of the most popular FCN architectures for medical image segmentation, which is a encoder-decoder network, but with additional short connection between encoder and decoder paths. Based on the fact that a pancreas only takes up a small fraction of the whole scan, Zhou *et al.* [30] proposed to find the rough pancreas region and then learn a FCN based fixed-point model to refine the pancreas region iteratively. Their method is also based on a coarse-to-fine framework, but it only considered coarse-to-fine RoIs. Besides coarse-to-fine RoIs, our coarse-to-fine method also takes coarse-to-fine overlap sizes into account, which is designed specifically for efficient 3D inference.

3.2.2 3D CNNs for Volumetric Segmentation

Although 2D CNNs based methods achieved considerable progresses, they are not optimal for medical image segmentation, as they cannot make full use of the 3D context encoded in volumetric data. Several 3D CNNs based segmentation methods have been proposed. The 3D U-Net [10] extended the previous 2D U-Net architecture [36] by replacing all 2D operations with their 3D counterparts. Based on the architecture of the 3D U-Net, the V-Net [11] introduced residual structures [61] (short term skip connection) into each stage of the network. Chen *et al* [33] proposed a deep voxel-wise residual network for 3D brain segmentation. Both I2I-3D [62] and 3D-DSN [37] included auxiliary supervision via side outputs into their 3D deep networks. Despite the success of 3D CNNs as a technique for segmenting the target organs, such as prostate [11] and kidney [10], very few techniques have been developed for leveraging 3D spatial information on the challenging pancreas segmentation. Gibson *et al.* [63] proposed the DenseVNet which is however constrained to have shallow encoders due to the computationally demanding dense connections. Roth *et al.* [91] extended 3D U-Net to segment pancreas, which has the following shortcomings, 1) the input of their networks is fixed to $120 \times 120 \times 120$, which is very computationally demanding

due to this large volume size, 2) the rough pancreas bounding box is resampled to a fixed size as their networks input, which loses information and flexibility, and cannot deal with the intrinsic large variations of pancreas in shape and size. Therefore, we propose our 3D coarse-to-fine framework that works on both normal and abnormal to ensure both low computation cost and high pancreas segmentation accuracy.

3.3 Method

In this section, we elaborate our proposed 3D coarse-to-fine framework which includes a ***coarse*** stage and a ***fine*** stage afterwards. We first formulate a segmentation model that can be generalized to both ***coarse*** stage and ***fine*** stage. Later in Sec. 3.3.1 and Sec. 3.3.2, we will customize the segmentation model to these two stages, separately.

We denote a 3D CT-scan volume by $\mathbf{X}$. This is associated with a human-labeled per-voxel annotation $\mathbf{Y}$, where both $\mathbf{X}$ and $\mathbf{Y}$ have size $W \times H \times D$, which corresponds to axial, sagittal and coronal views, separately. The ground-truth segmentation mask $\mathbf{Y}$ has a binary value y_i, $i = 1, \cdots, WHD$, at each spatial location i where $y_i = 1$ indicates that x_i is a pancreas voxel. Denote a segmentation model by $\mathbb{M} : \mathbf{P} = \mathbf{f}(\mathbf{X}; \boldsymbol{\Theta})$, where $\boldsymbol{\Theta}$ indicates model parameters and $\mathbf{P}$ means the binary prediction volume. Specifically in a neural network with L layers and parameters $\boldsymbol{\Theta} = \{\mathcal{W}, \mathcal{B}\}$, $\mathcal{W}$ is a set of weights and $\mathcal{B}$ is a set of biases, where $\mathcal{W} = \{\mathbf{W}^1, \mathbf{W}^2, \cdots, \mathbf{W}^L\}$ and $\mathcal{B} = \{\mathbf{B}^1, \mathbf{B}^2, \cdots, \mathbf{B}^L\}$. Given that $p(y_i|x_i; \boldsymbol{\Theta})$ represents the predicted probability of a voxel x_i being what is the labeled class at the final layer of the output, the negative log-likelihood loss can be formulated as:

$$\mathcal{L} = \mathcal{L}(\mathbf{X}; \boldsymbol{\Theta}) = -\sum_{x_i \in \mathbf{X}} \log(p(y_i|x_i; \boldsymbol{\Theta})). \tag{3.1}$$

It is also known as the cross entropy loss in our binary segmentation setting. By thresholding $p(y_i|x_i; \boldsymbol{\Theta})$, we can obtain the binary segmentation mask $\mathbf{P}$.

We also add some auxiliary layers to such a neural network (will be called mainstream network in the rest of the paper), which produces side outputs under deep supervision [92]. These auxiliary layers form a branch network and facilitate the feature learning at lower layer of the mainstream network. Each branch network shares the weights of the first d layers from the mainstream network, which is denoted by $\mathbf{\Theta}_d = \{\mathcal{W}_d, \mathcal{B}_d\}$. Apart from the shared weights, it owns weights $\widehat{\mathbf{\Theta}}_d$ to output the per-voxel prediction. Similarly, the loss of an auxiliary network can be formulated as:

$$\mathcal{L}_d(\mathbf{X}; \mathbf{\Theta}_d, \widehat{\mathbf{\Theta}}_d) = \sum_{x_i \in \mathbf{X}} -\log(p(y_i|x_i; \mathbf{\Theta}_d, \widehat{\mathbf{\Theta}}_d)), \tag{3.2}$$

which is abbreviated as $\mathcal{L}_d$. Finally, stochastic gradient descent is applied to minimize the negative log-likelihood, which is given by following regularized objective function:

$$\mathcal{L}_{overall} = \mathcal{L} + \sum_{d \in \mathcal{D}} \xi_d \mathcal{L}_d + \lambda(\|\mathbf{\Theta}\|^2 + \sum_{d \in \mathcal{D}} \|\widehat{\mathbf{\Theta}}_d\|)^2, \tag{3.3}$$

where $\mathcal{D}$ is a set of branch networks for auxiliary supervisions, ξ_d balances the importance of each auxiliary network and l_2 regularization is added to the objective to prevent the networks from overfitting. For conciseness concerns in the following sections, we keep a segmentation model that is obtained from the overall function described in Eq. 3.3 denoted by $\mathbb{M} : \mathbf{P} = \mathbf{f}(\mathbf{X}; \mathbf{\Theta})$, where $\mathbf{\Theta}$ includes parameters of the mainstream network and auxiliary networks.

3.3.1 Coarse Stage

In the ***coarse*** stage, the input of "ResDSN Coarse" is sampled from the whole CT-scan volume denoted by $\mathbf{X}^{\mathrm{C}}$, on which the ***coarse*** segmentation model $\mathbb{M}^{\mathrm{C}} : \mathbf{P}^{\mathrm{C}} = \mathbf{f}^{\mathrm{C}}\left(\mathbf{X}^{\mathrm{C}}; \mathbf{\Theta}^{\mathrm{C}}\right)$ is trained on. All the C superscripts depict the ***coarse*** stage. The goal of this stage is to efficiently produce a rough binary segmentation $\mathbf{P}^{\mathrm{C}}$ from the complex background, which can get rid of regions that are segmented as non-pancreas with a high confidence to obtain an approximate pancreas volume. Based on this approximate pancreas volume, we can crop from the original input $\mathbf{X}^{\mathrm{C}}$ with a rectangular cube

derived from $\mathbf{P}^{\mathrm{C}}$ to obtain a smaller 3D image space $\mathbf{X}^{\mathrm{F}}$, which is surrounded by simplified and less variable context compared with $\mathbf{X}^{\mathrm{C}}$. The mathematic definition of $\mathbf{X}^{\mathrm{F}}$ is formulated as:

$$\mathbf{X}^{\mathrm{F}} = \mathrm{Crop}[\mathbf{X}^{\mathrm{C}} \otimes \mathbf{P}^{\mathrm{C}}; \mathbf{P}^{\mathrm{C}}, m], \tag{3.4}$$

where $\otimes$ means an element-wise product. The function $\mathrm{Crop}[\mathbf{X}; \mathbf{P}, m]$ denotes cropping $\mathbf{X}$ via a rectangular cube that covers all the 1's voxels of a binary volume $\mathbf{P}$ added by a padding margin m along three axes. Given $\mathbf{P}$, the functional constraint imposed on $\mathbf{X}$ is that they have exactly the same dimensionality in 3D space. The padding parameter m is empirically determined in experiments, where it is used to better segment the boundary voxels of pancreas during the ***fine*** stage. The Crop operation acts as a dimensionality reduction to facilitate the fine segmentation, which is crucial to cut down the consuming time of segmentation. It is well-worth noting that the 3D locations of the rectangular cube which specifies where to crop $\mathbf{X}^{\mathrm{F}}$ from $\mathbf{X}^{\mathrm{C}}$ is recorded to map the ***fine*** segmentation results back their positions in the full CT scan.

3.3.2 Fine Stage

In the ***fine*** stage, the input of the ConvNet is sampled from the cropped volume $\mathbf{X}^{\mathrm{F}}$, on which we train the ***fine*** segmentation model $\mathbb{M}^{\mathrm{F}} : \mathbf{P}^{\mathrm{F}} = \mathbf{f}^{\mathrm{F}}\left(\mathbf{X}^{\mathrm{F}}; \mathbf{\Theta}^{\mathrm{F}}\right)$, where the F superscripts indicate the ***fine*** stage. The goal of this stage is to refine the coarse segmentation results from previous stage. In practice, $\mathbf{P}^{\mathrm{F}}$ has the same volumetric size of $\mathbf{X}^{\mathrm{F}}$, which is smaller than the original size of $\mathbf{X}^{\mathrm{C}}$.

3.3.3 Coarse-to-Fine Segmentation

Our segmentation task is to give a volumetric prediction on every voxel of $\mathbf{X}^{\mathrm{C}}$, so we need to map the $\mathbf{P}^{\mathrm{F}}$ back to exactly the same size of $\mathbf{X}^{\mathrm{C}}$ given by:

$$\mathbf{P}^{\mathrm{C2F}} = \mathrm{DeCrop}[\mathbf{P}^{\mathrm{F}} \odot \mathbf{P}^{\mathrm{C}}; \mathbf{X}^{\mathrm{F}}, \mathbf{X}^{\mathrm{C}}], \tag{3.5}$$

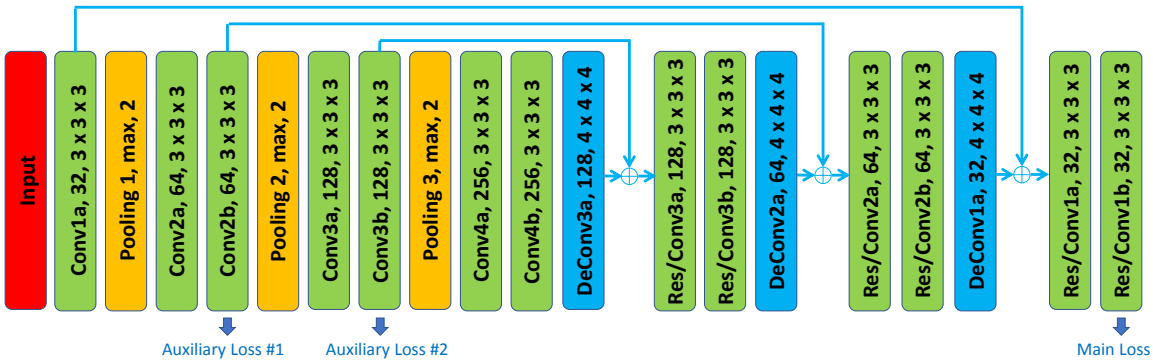

Figure 3.3. Illustration of our 3D convolutional neural network for volumetric segmentation. The encoder path is composed from "Conv1a" to "Conv4b" while the decoder path is from "DeConv3a" to "Res/Conv1b". Each convolution or deconvolution layer consists of one convolution followed by a BatchNorm and a ReLU. To clarify, "Conv1a, 32, $3 \times 3 \times 3$" means the convolution operation with 32 channels and a kernel size of $3 \times 3 \times 3$. "Pooling 1, max, 2" means the max pooling operation with kernel size of $2 \times 2 \times 2$ and a stride of two. Long residual connections are illustrated by the blue concrete lines. Blocks with same color mean the same operations. Best viewed in color.

where $\mathbf{P}^{\mathrm{C2F}}$ denotes the final volumetric segmentation, and $\odot$ means an element-wise replacement, and DeCrop operation defined on $\mathbf{P}^{\mathrm{F}}, \mathbf{P}^{\mathrm{C}}, \mathbf{X}^{\mathrm{F}}$ and $\mathbf{X}^{\mathrm{C}}$ is to replace a pre-defined rectangular cube inside $\mathbf{P}^{\mathrm{C}}$ by $\mathbf{P}^{\mathrm{F}}$, where the replacement locations are given by the definition of cropping $\mathbf{X}^{\mathrm{F}}$ from $\mathbf{X}^{\mathrm{C}}$ given in Eq. 3.4.

All in all, our entire 3D-based coarse-to-fine segmentation framework during testing is illustrated in Fig 3.2.

3.3.4 Network Architecture

As shown in Fig. 3.3, we provide an illustration of our convolutional network architecture. Inspired by V-Net [11], 3D U-Net [10], and VoxResNet [33], we have an encoder path followed by a decoder path each with four resolution steps. The left part of network acts as a feature extractor to learn higher and higher level of representations while the right part of network decompresses compact features into finer and finer resolution to predict the per-voxel segmentation. The padding and stride of each layer (Conv, Pooling, DeConv) are carefully designed to make sure the densely predicted

output is the same size as the input.

The encoder sub-network on the left is divided into different steps that work on different resolutions. Each step consists of one to two convolutions, where each convolution is composed of $3 \times 3 \times 3$ convolution followed by a batch normalization (BN [93]) and a rectified linear unit (ReLU [94]) to reach better convergence, and then a max pooling layer with a kernel size of $2 \times 2 \times 2$ and strides of two to reduce resolutions and learn more compact features. The downsampling operation implemented by max-pooling can reduce the size of the intermediate feature maps while increasing the size of the receptive fields. Having fewer size of activations makes it possible to double the number of channels during feature aggregation given the limited computational resource.

The decoder sub-network on the right is composed of several steps that operate on different resolutions as well. Each step has two convolutions with each one followed by a BatchNorm and a ReLU, and afterwards a Deconvolution with a kernel size of $4 \times 4 \times 4$ and strides of two is connected to expand the feature maps and finally predict the segmentation mask at the last layer. The upsampling operation that is carried out by deconvolution enlarges the resolution between each step, which increases the size of the intermediate activations so that we need to halve the number of channels due to the limited memory of the GPU card.

Apart from the left and right sub-networks, we impose a residual connection [61] to bridge short-cut connections of features between low-level layers and high-level layers. During the forward phase, the low-level cues extracted by networks are directly added to the high-level cues, which can help elaborate the fine-scaled segmentation, *e.g.*, small parts close to the boundary which may be ignored during the feature aggregation due to the large size of receptive field at high-level layers. As for the backward phase, the supervision cues at high-level layers can be back-propagated through the short-cut way via the residual connections. This type of mechanism can prevent networks from

Method	Long Res	Short Res	Deep Super	Loss
ResDSN (Ours)	Sum	No	Yes	CE
FResDSN	Sum	Sum	Yes	CE
SResDSN	No	Sum	Yes	CE
3D U-Net [10]	Concat	No	No	CE
V-Net [11]	Concat	Sum	No	DSC
VoxResNet [33]	No	Sum	Yes	CE
MixedResNet [96]	Sum	Sum	Yes	CE
3D DSN [37]	No	No	Yes	CE
3D HED [62]	Concat	No	Yes	CE

Table 3.1. Configurations comparison of different 3D segmentation networks on medical image analysis. For all the abbreviated phrases, "Long Res" means long residual connection, "Short Res" means short residual connection, "Deep Super" means deep supervision implemented by auxiliary loss layers, "Concat" means concatenation, "DSC" means Dice-Sørensen Coefficient and "CE" means cross-entropy. For residual connection, it has two types: concatenation ("Concat") or element-wise sum ("Sum").

gradient vanishing and exploding [95], which hampers network convergence during training.

We have one mainstream loss layer connected from "Res/Conv1b" and another two auxiliary loss layers connected from "Conv2b" and "Conv3b" to the ground truth label, respectively. For the mainstream loss in "Res/Conv1b" as the last layer which has the same size of data flow as one of the input, a $1 \times 1 \times 1$ convolution is followed to reduce the number of channels to the number of label classes which is 2 in our case. As for the two auxiliary loss layers, deconvolution layers are connected to upsample feature maps to be the same as input.

The deep supervision imposed by auxiliary losses provides robustness to hyper-parameters choice, in that the low-level layers are guided by the direct segmentation loss, leading to faster convergence rate. Throughout this work, we have two auxiliary branches where the default parameters are $\xi_1 = 0.2$ and $\xi_2 = 0.4$ in Eq. 3.3 to control the importance of deep supervisions compared with the major supervision from the mainstream loss for all segmentation networks.

As shown in Table 3.1, we give the detailed comparisons of network configurations

in terms of four aspects: long residual connection, short residual connection, deep supervision and loss function. Our backbone network architecture, named as "ResDSN", is proposed with different strategies in terms of combinations of long residual connection and short residual connection compared with VoxResNet [33], 3D HED [62], 3D DSN [37] and MixedResNet [96]. In this table, we also depict "FResDSN" and "SResDSN", where "FResDSN" and "SResDSN" are similar to MixedResNet [96] and VoxResNet [33], respectively. As confirmed by our quantitative experiments in Sec. 3.4.4.1, instead of adding short residual connections to the network, *e.g.*, "FResDSN" and "SResDSN", we only choose the long residual element-wise sum, which can be more computationally efficient while even performing better than the "FResDSN" architecture which is equipped with both long and short residual connections. Moreover, ResDSN has noticeable differences with respect to the V-Net [11] and 3D U-Net [10]. On the one hand, compared with 3D U-Net and V-Net which concatenate the lower-level local features to higher-level global features, we adopt the element-wise sum between these features, which outputs less number of channels for efficient computation. On the other hand, we introduce deep supervision via auxiliary losses into the network to yield better convergence.

3.4 Experiments

In this section, we first describe in detail how we conduct training and testing in the ***coarse*** and ***fine*** stages, respectively. Then we are going to compare our proposed method with previous state-of-the-art on two pancreas datasets: NIH pancreas dataset [18] and JHMI pathological pancreas dataset [88].

3.4.1 Network Training and Testing

All our experiments were run on a desktop equipped with the NVIDIA TITAN X (Pascal) GPU and deep neural networks were implemented based on the CAFFE [97]

platform customized to support 3D operations for all necessary layers, *e.g.*, "convolution", "deconvolution" and "pooling", *etc.* For the data pre-processing, we simply truncated the raw intensity values to be in $[-100, 240]$ and then normalized each raw CT case to have zero mean and unit variance to decrease the data variance caused by the physical processes [98] of medical images. As for the data augmentation in the training phase, unlike sophisticated processing used by others, *e.g.*, elastic deformation [11][36], we utilized simple but effective augmentations on all training patches, *i.e.*, rotation $(90°, 180°,$ and $270°)$ and flip in all three axes (axial, sagittal and coronal), to increase the number of 3D training samples which can alleviate the scarce of CT scans with expensive human annotations. Note that different CT cases have different physical resolutions, but we keep their resolutions unchanged. The input size of all our networks is denoted by $W_I \times H_I \times D_I$, where $W_I = H_I = D_I = 64$.

For the ***coarse*** stage, we randomly sampled $64 \times 64 \times 64$ sub-volumes from the whole CT scan in the training phase. In this case, a sub-volume can either cover a portion of pancreas voxels or be cropped from regions with non-pancreas voxels at all, which acts as a hard negative mining to reduce the false positive. In the testing phase, a sliding window was carried out to the whole CT volume with a ***coarse*** stepsize that has small overlaps within each neighboring sub-volume. Specifically, for a testing volume with a size of $W \times H \times D$, we have a total number of $(\lfloor \frac{W}{W_I} \rfloor + n) \times (\lfloor \frac{H}{H_I} \rfloor + n) \times (\lfloor \frac{D}{D_I} \rfloor + n)$ sub-volumes to be fed into the network and then combined to obtain the final prediction, where n is a parameter to control the sliding overlaps that a larger n results in a larger overlap and vice versa. In the ***coarse*** stage for the low time cost concern, we set $n = 6$ to efficiently locate the rough region of pancreas $\mathbf{X}^{\mathrm{F}}$ defined in Eq. 3.4 from the whole CT scan $\mathbf{X}^{\mathrm{C}}$.

For the ***fine*** stage, we randomly cropped $64 \times 64 \times 64$ sub-volumes constrained to be from the pancreas regions defined by ground-truth labels during training. In this case, a training sub-volume was assured to cover pancreatic voxels, which was

specifically designed to be capable of segmentation refinement. In the testing phase, we only applied the sliding window on $\mathbf{X}^{\mathrm{F}}$ with a size of $W_F \times H_F \times D_F$. The total number of sub-volumes to be tested is $(\lfloor \frac{W_F}{W_I} \rfloor + n) \times (\lfloor \frac{H_F}{H_I} \rfloor + n) \times (\lfloor \frac{D_F}{D_I} \rfloor + n)$. In the ***fine*** stage for the high accuracy performance concern, we set $n = 12$ to accurately estimate the pancreatic mask $\mathbf{P}^{\mathrm{F}}$ from the rough segmentation volume $\mathbf{X}^{\mathrm{F}}$. In the end, we mapped the $\mathbf{P}^{\mathrm{F}}$ back to $\mathbf{P}^{\mathrm{C}}$ to obtain $\mathbf{P}^{\mathrm{C2F}}$ for the final pancreas segmentation as given in Eq. 3.5, where the mapping location is given by the cropped location of $\mathbf{X}^{\mathrm{F}}$ from $\mathbf{X}^{\mathrm{C}}$.

After we get the final binary segmentation mask, we denote $\mathcal{P}$ and $\mathcal{Y}$ to be the set of pancreas voxels in the prediction and ground truth, separately, *i.e.*, $\mathcal{P} = \{i|p_i = 1\}$ and $\mathcal{Y} = \{i|y_i = 1\}$. The evaluation metric is defined by the Dice-Sørensen Coefficient (DSC) formulated as $\mathrm{DSC}(\mathcal{P}, \mathcal{Y}) = \frac{2\times|\mathcal{P}\cap\mathcal{Y}|}{|\mathcal{P}|+|\mathcal{Y}|}$. This evaluation measurement ranges in $[0, 1]$ where 1 means a perfect prediction.

3.4.2 NIH Pancreas Dataset

We conduct experiments on the NIH pancreas segmentation dataset [18], which contains 82 contrast-enhanced abdominal CT volumes provided by an experienced radiologist. The size of CT volumes is $512 \times 512 \times D$, where $D \in [181, 466]$ and their spatial resolutions are $w \times h \times d$, where $d = 1.0$mm and $w = h$ that ranges from 0.5mm to 1.0mm. Data pre-processing and data augmentation were described in Sec. 3.4.1. Note that we did not normalize the spatial resolution into the same one since we wanted to impose the networks to learn to deal with the variations between different volumetric cases. Following the training protocol [18], we perform 4-fold cross-validation in a random split from 82 patients for training and testing folds, where each testing fold has $21, 21, 20$ and 20 cases, respectively. We trained networks illustrated in Fig. 3.3 by SGD optimizer with a 16 mini-batch, a 0.9 momentum, a base learning rate to be 0.01 via polynomial decay (the power is 0.9) in a total of 80,000 iterations, and the weight

Method	Mean DSC	Max DSC	Min DSC
ResDSN C2F (Ours)	$\mathbf{84.59 \pm 4.86}\%$	$\mathbf{91.45}\%$	$\mathbf{69.62}\%$
ResDSN Coarse (Ours)	$83.18 \pm 6.02\%$	91.33%	58.44%
Cai *et al.* [35]	$82.4 \pm 6.7\%$	90.1%	60.0%
Zhou *et al.* [30]	$82.37 \pm 5.68\%$	90.85%	62.43%
Dou[1] *et al.* [37]	$82.25 \pm 5.91\%$	90.32%	62.53%
Roth *et al.* [29]	$78.01 \pm 8.20\%$	88.65%	34.11%
Yu[1] *et al.* [34]	$71.96 \pm 15.34\%$	89.27%	0%

Table 3.2. Evaluation of different methods on the NIH dataset. Our proposed framework achieves state-of-the-art by a large margin compared with previous state-of-the-arts.

decay 0.0005. Both training networks in the ***coarse*** and ***fine*** stages shared the same training parameter settings except that they took a $64 \times 64 \times 64$ input sampled from different underlying distributions described in Sec. 3.4.1, which included the details of testing settings as well. We average the score map of overlapped regions from the sliding window and throw away small isolated predictions whose portions are smaller than 0.2 of the total prediction, which can remove small false positives. For DSC evaluation, we report the average with standard deviation, max and min statistics over all 82 testing cases as shown in Table 3.2.

First of all, our overall coarse-to-fine framework outperforms previous state-of-the-art by nearly 2.2% (Cai *et al.* [35] and Zhou *et al.* [30]) in terms of average DSC, which is a large improvement. The lower standard deviation of DSC shows that our method is the most stable and robust across all different CT cases. Although the enhancement of max DSC of our framework is small due to the saturation, the improvement of min DSC over the second best (Dou *et al.* [37]) is from 62.53% to 69.62%, which is a more than 7% advancement. The worst case almost reaches 70%, which is a reasonable and acceptable segmentation result. After coarse-to-fine, the segmentation result of the worst case is improved by more than 11% after the 3D-based refinement from the 3D-based coarse result. The overall average DSC was also improved by 1.41%, which

[1]The results are reported by our runs using the same cross-validation splits where code is available from their GitHub: https://github.com/yulequan/HeartSeg.

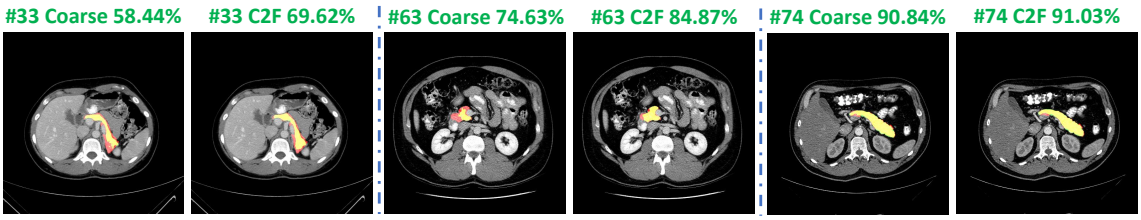

Figure 3.4. Examples of segmentation results reported by "ResDSN Coarse" and "ResDSN C2F" on a same slice in the axial view from NIH case #33, #63 and #74, respectively. Numbers after "Coarse" or "C2F" mean testing DSC. Red, green and yellow indicate the ground truth, prediction and overlapped regions, respectively. Best viewed in color.

proves the effectiveness of our framework.

As shown in Fig 3.4, we report the segmentation results by "ResDSN Coarse" and "ResDSN C2F" on the same slice for comparison. Note that yellow regions are the correctly predicted pancreas. For the NIH case #33, which is the min DSC case reported by both "ResDSN Coarse" and "ResDSN C2F", the "ResDSN C2F" successfully predict more correct pancreas regions at the bottom, which is obviously missed by "ResDSN Coarse". If the coarse segmentation is bad, *e.g.*, case #33 and #63, our 3D coarse-to-fine can significantly improve the segmentation results by as much as 10% in DSC. However, if the coarse segmentation is already very good, *e.g.*, case #74, our proposed method cannot improve too much. We conclude that our proposed "ResDSN C2F" shows its advancement over 2D methods by aggregating rich spatial information and is more powerful than other 3D methods on the challenging pancreas segmentation task.

3.4.3 JHMI Pathological Pancreas Dataset

We verified our proposed idea on the JHMI pathological cyst dataset [88] of abdominal CT scans as well. Different from the NIH healthy pancreas dataset, this dataset includes pathological cysts where some can be or can become cancerous. The pancreatic cancer stage largely influences the morphology of the pancreas [99] that makes this dataset extremely challenging for considering the large variants.

Method	Mean DSC
ResDSN C2F (Ours)	$\mathbf{80.56 \pm 13.36}\%$
ResDSN Coarse (Ours)	$77.96 \pm 13.36\%$
Zhou *et al.* [88]	$79.23 \pm 9.72\%$

Table 3.3. Evaluations on the JHMI pathological pancreas.

This dataset has a total number of 131 contrast-enhanced abdominal CT volumes with human-labeled pancreas annotations. The size of CT volumes is $512 \times 512 \times D$, where $D \in [358, 1121]$ that spans a wider variety of thickness than one of the NIH dataset. Following the training protocol [88], we conducted 4-fold cross validation on this dataset where each testing fold has $33, 33, 32$ and 33 cases, respectively. We trained networks illustrated in Fig. 3.3 in both the ***coarse*** and ***fine*** stage with the same training settings as on the NIH except that we trained a total of 300,000 iterations on this pathological dataset since a pancreas with cysts is more difficult to segment than a normal case. In the testing phase, we vote the prediction map of overlapped regions from the sliding window and ignore small isolated pancreas predictions whose portions are smaller than 0.05 of the total prediction. As shown in Table. 3.3, we compare our framework with only one available published results on this dataset. "ResDSN C2F" achieves an average 80.56% DSC that consistently outperforms the 2D based coarse-to-fine method [88], which confirms the advantage of leveraging the rich spatial information along three axes. What's more, the "ResDSN C2F" improves the "ResDSN Coarse" by 2.60% in terms of the mean DSC, which is a remarkable improvement that proves the effectiveness of the proposed 3D coarse-to-fine framework. Both [88] and our method have multiple failure cases whose testing DSC are 0, which indicates the segmentation of pathological organs is a more tough task. Due to these failure cases, we observe a large deviation on this pathological pancreas dataset compared with results on the NIH healthy pancreas dataset.

Method	Mean DSC	Max DSC	Min DSC
ResDSN Coarse (Ours)	$\mathbf{83.18 \pm 6.02}\%$	91.33%	58.44%
FResDSN Coarse	$83.11 \pm 6.53\%$	91.34%	61.97%
SResDSN Coarse	$82.82 \pm 5.97\%$	90.33%	62.43%
DSN [37] Coarse	$82.25 \pm 5.91\%$	90.32%	62.53%

Table 3.4. Evaluation of different residual connections on NIH.

3.4.4 Discussion

In this section, we conduct the ablation studies about residual connection, time efficiency and deep supervision to further investigate the effectiveness and efficiency of our proposed framework for pancreas segmentation.

3.4.4.1 Residual Connection

We discuss how different combinations of residual connections contribute to the pancreas segmentation task on the NIH dataset. All the residual connections are implemented in the element-wise sum and they shared exactly the same deep supervision connections, cross-validation splits, data input, training and testing settings except that the residual structure is different from each other. As given in Table. 3.4, we compare four configurations of residual connections of 3D based networks only in the ***coarse*** stage. The major differences between our backbone network "ResDSN" with respect to "FResDSN", "SResDSN" and "DSN" are depicted in Table. 3.1. "ResDSN" outperforms other network architectures in terms of average DSC and a small standard deviation even through the network is not as sophisticated as "FResDSN", which is the reason we adopt "ResDSN" for efficiency concerns in the ***coarse*** stage.

3.4.4.2 Time Efficiency

We discuss the time efficiency of the proposed coarse-to-fine framework with a smaller overlap in the ***coarse*** stage for the low consuming time concern while a larger one in the ***fine*** stage for the high prediction accuracy concern. The overlap size depends

Method	Mean DSC	n	Testing Time (s)
ResDSN C2F (Ours)	$\mathbf{84.59 \pm 4.86}\%$	6&12	245
ResDSN Coarse (Ours)	$83.18 \pm 6.02\%$	6	**111**
ResDSN Fine (Ours)	$83.96 \pm 5.65\%$	12	382

Table 3.5. Average time cost in the testing phase, where n controls the overlap size of sliding windows during the inference.

on how large we choose n defined in Sec 3.4.1. We choose $n = 6$ during the coarse stage while $n = 12$ during the fine stage. Experimental results are shown in Table 3.5. "ResDSN Coarse" is the most efficient while the accuracy is the worst among three methods, which makes sense that we care more of the efficiency to obtain a rough pancreas segmentation. "ResDSN Fine" is to use a large overlap on an entire CT scan to do the segmentation which is the most time-consuming. In our coarse-to-fine framework, we combine the two advantages together to propose "ResDSN C2F" which can achieve the best segmentation results while the average testing time cost for each case is reduced by 36% from 382s to 245s compared with "ResDSN Fine". In comparison, it takes an experienced board certified Abdominal Radiologist 20 mins for one case, which verifies the clinical use of our framework.

3.4.4.3 Deep Supervision

We discuss how effective of the auxiliary losses to demonstrate the impact of the deep supervision on our 3D coarse-to-fine framework. Basically, we train our mainstream networks without any auxiliary losses for both coarse and fine stages, denoted as "Res C2F", while keeping all other settings as the same, *e.g.*, cross-validation splits, data pre-processing and post-processing. As shown in Table 3.6, "ResDSN C2F" outperforms "Res C2F" by 17.79% to a large extent on min DSC and 0.53% better on average DSC though it's a little bit worse on max DSC. We conclude that 3D coarse-to-fine with deep supervisions performs better and especially more stable on the pancreas segmentation.

Method	Mean DSC	Max DSC	Min DSC
ResDSN C2F (Ours)	$\mathbf{84.59 \pm 4.86}\%$	91.45%	**69.62**%
Res C2F	$84.06 \pm 6.51\%$	**91.72**%	51.83%

Table 3.6. Discussions of the deep supervision on NIH.

3.5 Conclusion and Future Works

In this work, we propose a novel 3D network called "ResDSN" integrated with a coarse-to-fine framework to simultaneously achieve high segmentation accuracy and low time cost. The backbone network "ResDSN" is carefully designed to only have long residual connections for efficient inference. To our best knowledge, we are one of the first works to segment the challenging pancreas using 3D networks which leverage the rich spatial information to achieve the state-of-the-art. On widely-used datasets, the worst segmentation case is experimentally improved a lot by our coarse-to-fine framework. What's more, our coarse-to-fine framework can work on both normal and abnormal pancreases to achieve good segmentation accuracy. As future works, there are two directions to explore. First, the network backbone can be further advanced by the automatically searching the neural network architectures. Second, current coarse-to-fine framework decouples the coarse and fine stages, whereas jointly optimizing the segmentation networks in these two stages can further boost the overall segmentation accuracy.

Chapter 4

Multi-Scale Coarse-to-Fine Segmentation for Screening Pancreatic Ductal Adenocarcinoma

In this chapter, we propose an intuitive approach of detecting pancreatic ductal adenocarcinoma (PDAC), the most common type of pancreatic cancer, by checking abdominal CT scans. Our idea is named **multi-scale segmentation-for-classification**, which classifies volumes by checking if at least a sufficient number of voxels is segmented as tumors, by which we can provide radiologists with tumor locations. In order to deal with tumors with different scales, we train and test our volumetric segmentation networks with multi-scale inputs in a coarse-to-fine flowchart. A post-processing module is used to filter out outliers and reduce false alarms. We collect a new dataset containing 439 CT scans, in which 136 cases were diagnosed with PDAC and 303 cases are normal, which is the largest set for PDAC tumors to the best of our knowledge. To offer the best trade-off between sensitivity and specificity, our proposed framework reports a sensitivity of 94.1% at a specificity of 98.5%, which demonstrates the potential to make a clinical impact.

4.1 Introduction

Pancreatic cancer is one of the most dangerous killers to human lives, causing more than 330,000 deaths globally in 2014 [22]. Pancreatic ductal adenocarcinoma (PDAC) is the most common type of pancreatic cancer, accounting for about 85% of cancer cases. In early stages, this disease often has few symptoms and is very difficult to discover. By the time of diagnosis, the cancer has often spread to other parts of the body, leading to a very poor prognosis (*e.g.*, a five-year survival rate of 5% [22]). But, for cases diagnosed early, the survival rate rises to about 20% [23]. Hence, it is very important to study the possibility of detecting PDAC in common examinations, *e.g.*, the abdominal CT scan.

The early diagnosis of pancreatic cancer requires much expertise in reading the scanned images and making decisions, but the increasing number of cases makes it impossible for a limited number of experienced radiologists to check all CT scans manually. Therefore, an artificial intelligence system for this purpose is in need. In particular, the radiologists in our team are interested in a system working on abdominal CT scans, which filters out a large fraction of normal cases, but preserves almost all abnormal cases for further investigation. To the best of our knowledge, there is no existing work on this task.

With the development of deep learning [59], it is possible to construct a system which learns from professional knowledge in data annotation, and apply it to helping doctors in various clinical purposes. The pancreas is one of the most challenging organs in CT segmentation [18]. The difficulty mainly lies in its irregular shape and low contrast around the boundary. Powered by the recent progress in deep learning for 2D [26][36] and 3D [11][64] image segmentation, researchers designed various approaches [29][38] towards accurate pancreas segmentation. In the pathological cases, the morphology of the pancreas can be largely impacted by the difference in the

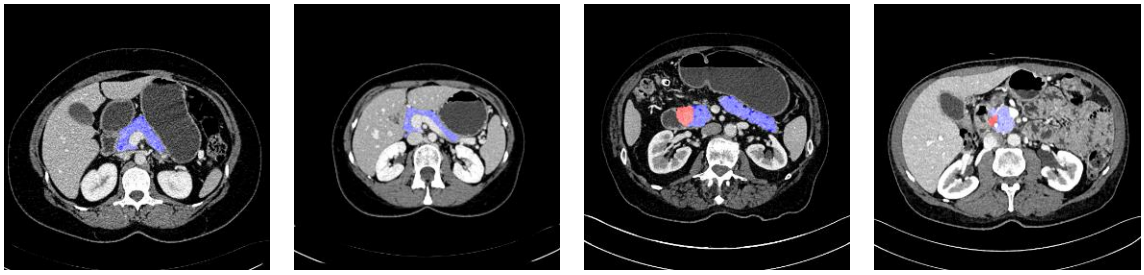

Figure 4.1. Examples of normal and abnormal (PDAC) pancreases (best viewed in color). Blue and red region mark the normal pancreas and tumor regions, respectively.

pancreatic cancer stage [100][88].

Our work is aimed to detect PDAC from a mixture of normal and abnormal CT scans. This is not a simple classification since radiologists also want to know the location of PDAC, we suggest a solution named **segmentation-for-classification**, which trains segmentation models and uses their outputs for classification. To deal with tumors of various sizes (Fig. 4.1), we adopt a segmentation network with multiple input scales, *i.e.*, 64^3, 32^3 and 16^3 volumes. But, voting that small input regions lead to a high false alarm rate, we adopt a **coarse-to-fine** testing strategy, which uses the 64^3 network for a coarse scan, and then the 32^3&16^3 networks inside the bounding box to detect small tumors that are possibly ignored in the previous stage. A non-parameterized post-processing algorithm is designed to remove outliers.

Our contributions are three folds: 1) we voxelwisely annotate an abdominal CT dataset with 439 cases in total, in which 136 cases are diagnosed with PDAC while the remaining 303 cases are normal, which is currently the largest PDAC dataset to the best of our knowledge; 2) we adopt a **multi-scale segmentation-for-classification** framework to conduct an **interpretable** abnormality detection, which provides radiologists with suspicious regions for further diagnosis; 3) our framework achieves a sensitivity of 94.1% at a specificity of 98.5%, which shows a promising direction to make a potential significant clinical impact.

4.2 Method

4.2.1 The Overall Framework

Let a dataset be $\mathbf{S} = \{(\mathbf{X}_1, y_1^\star), \ldots, (\mathbf{X}_N, y_N^\star)\}$, where N is the number of CT scans, $\mathbf{X}_n \in \mathbb{R}^{W_n \times H_n \times L_n}$ is the 3D volume with each element indicating the Hounsfield unit (HU) of a voxel, and $y_n \in \{0, 1\}$ is the label (0 for a normal case, 1 for an abnormal case). Throughout this paper, by *abnormal* we refer to the cases diagnosed as PDAC. The goal is to design a model $\mathbb{M} : y = f(\mathbf{X})$ to predict the label for each testing volume. We evaluate our approach by ranking all volumes by the probability of being a PDAC, computing the sensitivity and specificity at a given threshold, and plotting the ROC curve indicating the relationship between the sensitivity and specificity at different thresholds. For clinical purposes, we shall guarantee a high sensitivity with a reasonable specificity.

Although some previous work suggested to classify CT or MRI volumes directly using 3D networks [101][102], we argue that a better solution is to perform tumor segmentation at the same time of classification. This makes the classification results **interpretable** by segmentation cues, by which radiologists can take a further investigation of the suspicious abnormal regions. In addition, this integrates voxel-wise annotations into the classification model as deep supervision, so that the entire network is better trained [88]. Therefore, we propose a two-stage framework named *segmentation-for-classification*, in which a segmentation stage first extracts voxel-wise cues from the input CT scan, and a classification stage follows to summarize this information into the final prediction. Our multi-scale segmentation strategy is different from [38], which applied another network of the same scale in the fine stage. **Tumor detection requires multiple scales.**

Mathematically, let each training data be augmented by a segmentation mask $\mathbf{M}_n^\star$ of the same dimensionality as $\mathbf{X}$, so that $m_{n,i}^\star \in \{0, 1, 2\}$ indicates the category of the

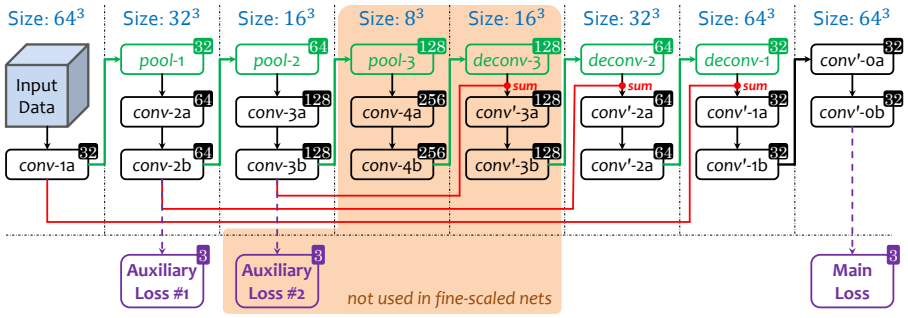

Figure 4.2. The architecture of our segmentation backbone (best viewed in color). Each rectangle is a layer, green arrows indicate operations changing spatial resolution, and red arrows mean residual connections. We illustrate the situation when the input volume size is 64^3. If it is 32^3 or 16^3, all volumes are shrunk accordingly (to $1/2^3$ or $1/4^3$ of the displayed size). The number at the upper-right corner of each cube is the number of channels. Each convolution uses $3\times3\times3$ kernels with 1 as stride, each pooling $2\times2\times2$ with 2 as stride, and each deconvolution $4\times4\times4$ with 2 as stride. The weight ratio for auxiliary losses $\#1$, $\#2$ and main loss is $1:2:5$ for the 64^3 network, and $1:3$ for the auxiliary loss $\#1$ and the main loss for the 32^3 and 16^3 networks.

i-th voxel, *i.e.*, in the tumor ($m_{n,i}=2$), outside the tumor but inside the pancreas ($m_{n,i}=1$), or outside the pancreas ($m_{n,i}=0$). Note that the tumor voxel set is a subset of the pancreas voxel set. The segmentation module is a high-dimensional function $\mathbf{M}=\mathbf{s}(\mathbf{X})$, which is implemented by a deep encoder-decoder network. The classification module is a binary function $y=c(\mathbf{M})$. The overall framework is thus written as:

$$y=f(\mathbf{X})=c\circ\mathbf{s}(\mathbf{X}). \tag{4.1}$$

4.2.2 Training: Multi-Scale Deeply-Supervised Segmentation

We start with describing the segmentation stage. The tumor region in a pancreas, as shown in Fig. 4.1, can vary in scale, appearance and geometric properties. In particular, the largest tumor in our dataset occupies over one million voxels, but the smallest one has only thousands. This motivates us to train multi-scale networks to deal with such a large variation in scale.

In practice, we train three networks, taking input volumes of 64^3, 32^3 and 16^3 voxels,

respectively. Each segmentation network follows an encoder-decoder flowchart shown in Fig. 4.2. It has a series of convolutional layers to learn 3D patterns from training data. Down-sampling and up-sampling are implemented by max pooling and deconvolutional layers, respectively. Following [38], we introduce deep supervision in the training process, which is implemented by adding several auxiliary losses to intermediate layers, which delivers better performance for the normal and cystic pancreas segmentation in [38]. Deep supervision is considered as a way of incorporating multi-stage visual cues, which constrains intermediate layers and improves the stability of training deep networks. Multi-scale segmentation is complementary to deep supervision, which aims at capturing visual patterns of various scales. As can be seen in experiments, multi-scale segmentation can take advantage of different scales, *i.e.*, a large network produces a high specificity, and a small network gives a high sensitivity.

The training process starts with sampling patches of a specified size. Since the pancreas and the tumor only occupy a small fraction of the entire volume, a random sampling strategy may lead to that only few patches contain pancreas or tumor voxels, and thus the segmentation models are biased towards the background class. To deal with the issue, we sample lots of foreground patches for training the 32^3 and 16^3 networks. We first compute the region-of-interest (ROI) by padding a 32-voxel margin around the minimal 3D bounding box covering the entire pancreas. Within it, we categorize the randomly sampled patches into three types (*i.e.*, *background*, *tumor* and *pancreas*) according to the fraction of pancreas and tumor voxels, and make the numbers of training patches of these types to be approximately the same. Data augmentation is performed by randomly flipping patches and rotating by 90°, 180° and 270° over three axes.

We use the same configuration for training these networks. The base learning rate is 0.01 and decayed polynomially (the power is 0.9) in a total of 80,000 iterations (the mini-batch size is 16, 32 and 128 for 64^3, 32^3 and 16^3, respectively). The weight decay

and momentum are set to be 0.0005 and 0.9, respectively.

4.2.3 Testing: Coarse-to-Fine Segmentation

The first goal is to perform the pancreas and tumor segmentation. We first slide a 64^3 window in the entire CT volume. The spatial stride is 20 along three axes, which is chosen to have the average testing time for each case within 11 minutes on a TITAN Xp GPU. Based on the *coarse* segmentation, we compute the ROI, *i.e.*, the smallest box covering all pancreas and tumor voxels padded by 32, and crop the CT image accordingly. Then, we scan the ROI with sliding windows of 32^3 and 16^3 voxels, and the strides are set to be 10 and 5, respectively. We do not run the two small networks on the entire volume because it can easily hallucinate tumors in the background regions. In addition, shrinking the scanning region for the 32^3 and 16^3 networks leads to a significant speedup in the testing process. The predictions of three networks are averaged into final segmentation.

Then, based on the segmentation mask, we classify each volume as normal or abnormal. Advised by the radiologists who desire the classification result to be explainable, we do not formulate the classifier $c(\cdot)$ as another deep network, but use a simple, non-parametrized approach to filter out the outliers. We construct a graph on all voxels predicted as *normal pancreas* or *tumor*. Each voxel is a node, and there exists an edge between the adjacent voxels (each voxel is adjacent to 6 neighbors). We compute all connected component in the graph. A component is preserved if it is larger than 20% of the maximal connected component, otherwise it is removed, *i.e.*, all voxels within this component are predicted as *background*. To obtain our final goal, a volume is predicted as PDAC if at least K voxels are predicted as tumor. In practice, we empirically set $K = 50$.

4.3 Experiments

4.3.1 Dataset and Settings

We collected a dataset with 303 normal cases from potential renal donors, as well as 136 biopsy-proven PDAC cases. Four experts in abdominal anatomy annotated the pancreas and tumor voxels on these data using the Varian Velocity software, and each case was checked by an experienced board-certified Abdominal Radiologist. For a radiologist, an average normal case took 20 minutes, and an average abnormal case 40 minutes to segment. Since the abnormal cases are much harder to obtain and annotate than the normal cases, we adopt a 4-fold cross-validation on our 136 PDAC scans to have testing results on every abnormal case while we use a hard split of training and testing on our 303 normal cases. All in all, each training set contains 103 normal and 102 abnormal cases where the normal-to-abnormal ratio is close to 1, and each testing set contains 34 abnormal and 200 normal cases. The average size of CT scans is 512×512×667.

One goal is to measure the segmentation accuracy by the Dice-Sørensen Coefficient (DSC) between the predicted and the ground-truth tumor sets $\mathcal{Y}$ and $\mathcal{Y}^\star$, *i.e.*, $\mathrm{DSC}(\mathcal{Y}, \mathcal{Y}^\star) = 2 \times |\mathcal{Y} \cap \mathcal{Y}^\star| / (|\mathcal{Y}| + |\mathcal{Y}^\star|)$. Our main goal is the tumor classification, which involves a tradeoff between sensitivity and specificity.

4.3.2 Segmentation Results

We first summarize the segmentation results in Table 4.1, which makes the normal v.s. abnormal classification to be interpretable by segmentation cues. The 64^3 network achieves reasonable pancreas and tumor segmentation accuracies. The segmentation result of normal pancreas is as high as 86.9%, which means that the normal pancreases are easier to segment, as there are often unpredicted changes in shape and geometry in the abnormal cases. As a side comment, the lowest DSC of an abnormal pancreas is

Scale	N. Pancreas	A. Pancreas	Tumor	Misses	Sen	Spe
64^3	$\mathbf{86.9} \pm 8.6\%$	$\mathbf{81.0} \pm 10.8\%$	$\mathbf{57.3} \pm 28.1\%$	10/136	92.7%	**99.0**%
32^3	$82.0 \pm 12.2\%$	$75.7 \pm 14.9\%$	$53.8 \pm 26.1\%$	7/136	94.9%	96.0%
16^3	$61.5 \pm 20.6\%$	$64.1 \pm 20.2\%$	$42.5 \pm 25.6\%$	**4**/136	**97.1**%	86.5%
Multi	$84.5 \pm 11.1\%$	$78.6 \pm 13.3\%$	$56.5 \pm 27.2\%$	8/136	94.1%	98.5%

Table 4.1. Comparison of segmentation and classification results by networks of different scales and their combination. From left to right: normal/abnormal pancreas and tumor segmentation accuracy (DSC, %), the number of missing tumors (*i.e.*, DSC is 0%), and the sensitivity (abbreviated as sen, and sen $= 1 -$ miss rate) and specificity (abbreviated as spe).

38.4%, lower than the number (44.0%) of a normal pancreas. In tumor segmentation, we observe a lower accuracy and a higher standard deviation ($57.3 \pm 28.1\%$). Except for the 10 missing cases, we find 20 more cases with a tumor DSC lower than 30%. All these evidences imply the challenging of finding tumors considering their various size, shape and locations. Note that a recent work on the pancreatic cyst segmentation achieves a DSC of $63.4 \pm 27.7\%$ [88], which is not as hard as the tumor segmentation.

Going to smaller scales, fewer tumors are missed, though segmentation accuracies become lower. This is the tradeoff between sensitivity and specificity: a network with a smaller input region has the ability to detect tiny regions, but without seeing contexts, it can be easily confused by false positives. Thus, combining multi-scale predictions achieves a balance between sensitivity and specificity. Fig. 8.3 shows two examples that benefit from multi-scale segmentation.

We replace our backbone with 3D UNet [10] and VNet [11] at the 64^3 scale setting and report their results in Table 4.2 for comparison. We can find that the three backbones perform roughly similar in terms of the segmentation results. However, our backbone achieves the best results for the sensitivity and specificity.

Method	N. Pancreas	A. Pancreas	Tumor	Misses	Sen	Spe
Ours	$86.9 \pm 8.6\%$	$81.0 \pm 10.8\%$	$57.3 \pm 28.1\%$	**10/136**	**92.7%**	**99.0%**
UNet	$\mathbf{87.0 \pm 8.4\%}$	$\mathbf{81.6 \pm 10.2\%}$	$57.6 \pm 27.8\%$	11/136	91.9%	99.0%
VNet	$86.7 \pm 8.8\%$	$80.6 \pm 11.4\%$	$\mathbf{58.7 \pm 28.0\%}$	10/136	92.7%	98.0%

Table 4.2. Comparison of different networks as backbone at the 64^3 setting. The sensitivity is abbreviated as sen and the specificity is abbreviated as spe.

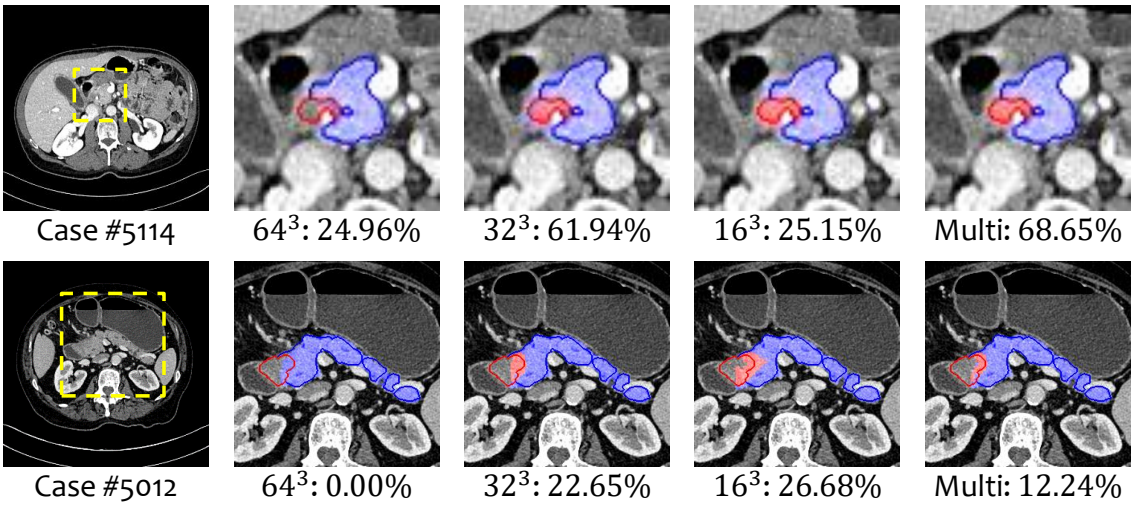

Figure 4.3. Multi-scale segmentation examples (best viewed in color). Top: a case that all three scales work well, and multi-scale combines them to achieve a higher DSC. Bottom: a failure case in the 64^3 network, but found by the 32^3 and 16^3 networks. The yellow frames indicate the zoomed-in regions, the blue and red contours mark the annotated pancreas and tumor respectively, and the masked regions mark segmentation results.

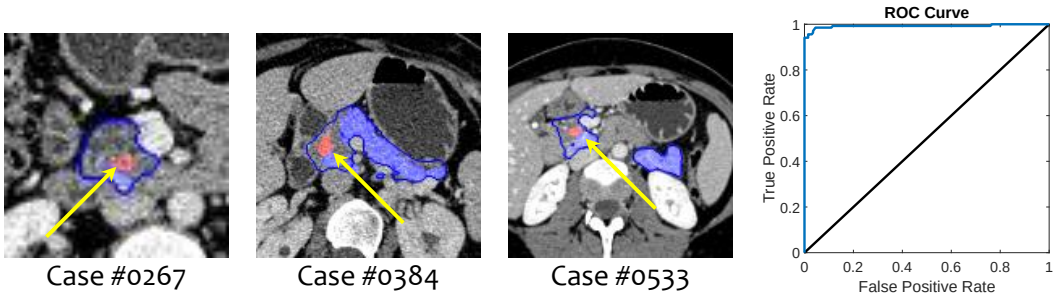

Figure 4.4. Left: three false alarm examples, in which the blue contour marks the annotated pancreas, and the blue and red regions mark the predicted pancreas and tumor, respectively. We use yellow arrows to indicate the detected tiny "tumors". Right: the ROC curve of multi-scale classification. This figure is best viewed in color.

4.3.3 Classification Results

Finally, we summarize classification results in Table 4.1, which is the crucial goal of making earlier diagnosis possible for doctors. Radiologists care more about a high sensitivity since they don't want to miss a patient who has an abnormal pancreas, which inspires us to adopt a multi-scale strategy to improve the sensitivity while keeping a reasonable specificity. The model with multi-scale information achieves the best overall performance, *i.e.*, a sensitivity of 94.1% at a specificity of 98.5%. These high scores imply that tumor segmentation provide strong cues for PDAC screening. We show all three false alarms in Fig. 4.4. The radiologists of our team confirmed that 2 out of these 3 false positives have focal fatty infiltration in the pancreas corresponding to the detected "tumors". Focal fatty infiltration is difficult for radiologists to distinguish from tumor in current clinical practice. In this case, the predicted "false alarm" was not normal in view of our radiologists.

By augmenting our segmentation for classification framework with cues from number of predicted tumor voxels since the more voxels predicted as PDAC the more likely this case is abnormal, we can output a confident score for each case, indicating the possibility that this case suffers PDAC. More specifically, a confidence score is computed by a weighted sum of the volume size and the segmentation probability of predicted tumor voxels. By sorting all testing cases according to their confident scores, we obtain a ROC curve of sensitivity and specificity. From the ROC curve, we can make different emphasis to change the tradeoff between sensitivity and specificity, *e.g.*, we can achieve a sensitivity of 98.5% at a specificity of 95.6%, or a specificity of 99.5% at a sensitivity of 94.1%.

4.4 Conclusion and Future Works

In this chapter, we study an important and challenging task, *i.e.*, detecting pancreases diagnosed with PDAC in abdominal CT scans. This topic is crucial in saving lives from pancreatic cancer yet few studied before, possibly due to the lack of data. We propose a **segmentation-for-classification** framework which trains a segmentation network and performs **interpretable** abnormality classification by simply checking the existence of tumor voxels in each testing volume. There are two key points to improve classification accuracy, known as **multi-scale** network training and **coarse-to-fine** testing. To offer a best trade-off between sensitivity and specificity on our own collected dataset containing 303 normal and 136 PDAC cases, we achieve a sensitivity of 94.1% at a specificity of 98.5%. The strong numbers show the promising direction to make a significant impact in clinics for early detection of pancreatic cancer, which would save lives. Future works will include the dual-phase information (arterial and venous phases) since current segmentation for classification is only done on venous phase. Another possible direction is to label and segment the dilated pancreatic duct in the meantime since it is regarded as the sign of high risk for pancreatic cancers considering the dilated pancreatic duct sometimes can be easier to locate than PDAC.

Chapter 5

Segmentation for Classification of Screening Pancreatic Neuroendocrine Tumors

This work presents comprehensive results to detect in the early stage the pancreatic neuroendocrine tumors (PNETs), a group of endocrine tumors arising in the pancreas, which are the second common type of pancreatic cancer, by checking the abdominal CT scans. To the best of our knowledge, this task has not been studied before as a computational task. To provide radiologists with tumor locations, we adopt a segmentation framework to classify CT volumes by checking if at least a sufficient number of voxels is segmented as tumors. To quantitatively analyze our method, we collect and voxelwisely label a new abdominal CT dataset containing 376 cases with both arterial and venous phases available for each case, in which 228 cases were diagnosed with PNETs while the remaining 148 cases are normal, which is currently the largest dataset for PNETs to the best of our knowledge. In order to incorporate rich knowledge of radiologists to our framework, we annotate dilated pancreatic duct as well, which is regarded as the sign of high risk for pancreatic cancer. Quantitatively, our approach outperforms state-of-the-art segmentation networks and achieves a sensitivity of 89.47% at a specificity of 81.08%, which indicates a potential direction to achieve a clinical impact related to cancer diagnosis by earlier tumor detection.

5.1 Introduction

The American Cancer Society estimates that about 56,770 people in the United States will be diagnosed with pancreatic cancer in 2019, and that 45,750 will die from the disease [103]. The oncology community has expended arsenal at this disease with little effect: the 5-year survival rate remains at only $\approx 5\%$ [22] despite decades of effort. This is due to the fact that most patients with localized disease have no recognizable symptoms or signs; as a result, upon diagnosis, tumors have generally spread to critical abdominal vessels and/or adjacent organs, which is too late to be cured. Despite the grim statistics, there is still real hope for the early detection, which can boost the 5-year survival rate by 3 times to reach around 20% given an early diagnosis [23]. Among pancreatic cancers, the pancreatic adenocarcinoma (PDAC) is the most common type of pancreatic cancer. Recently, there is a study [20] showing that they can achieve an overall sensitivity of 94.1% at a specificity of 98.5% for the detection of PDAC, which sheds light on the possibility of early pancreatic cancer detection. In our work, we focus on the early detection of pancreatic neuroendocrine tumors (PNETs) from CT scans, which is harder than the detection of PDAC considering PNETs are less common with even smaller voxel size.

The detection of PNETs falls into the area of computer aided diagnosis. The main challenges are lying in three folds: 1) the small size of tumors with respect to the whole volume; 2) the large tumor variations in location, shape and size across different patients; 3) the abnormalities can change the texture of surrounding tissues a lot, which makes the task even harder to locate the tumor targets. With the unprecedented development of deep learning, in particularly fully convolutional neural networks (FCNs), there are works which has been driving the field forward in image segmentation [11], [26], [36]. In the pancreas segmentation area, researchers have been actively pushing the boundaries of obtaining accurate segmentation performance on

both normal pancreas [29], [35] and abnormal pancreas [38], [88].

Valuable insights from radiologists' clinical diagnosis and analysis process can be leveraged to tackle this problem. **First**, radiologists are very sensitive to the dilated pancreatic duct when reading CT scans. There are often occasions the pancreatic duct is visible to be dilated though the PNETs are barely visible from CT appearance and texture. **Second**, the appearance and texture cues can be very different for PNETs between the venous and arterial phases of CT scans. Radiologists make diagnosis decision by checking both phases in case the PNETs are hardly to be picked up in one phase. Missing true PNETs can cause critical areas to remain untreated. To migrate the aforementioned practical knowledge from radiologists routine works to our system, on the one hand, we annotated the voxels of dilated pancreatic duct as strong auxiliary cues to indicate the present of pancreatic cancer. On the other hand, we conduct PNETs segmentation and classification on both arterial and venous phases to reduce the missing detection of PNETs. This is done quite different from the state-of-the-art work on PDACs [20], where they only study on one phase and no cues are explored from the dilated pancreatic duct. Our final goal is aimed to detect PNETs from a mixed set of normal and abnormal CT scans. It is not a simple binary classification task because radiologists want to know the location of tumors, so we use the idea of Segmentation-for-Classification (S4C), which trains segmentation models and uses voxel predictions for the classification.

Our **contributions** are three folds: 1) we voxelwisely label a dual-phase PNETs dataset in both arterial and venous phases, which is the largest dataset and study up-to-date to the best of our knowledge; 2) we are the first work of segmentation and classification for PNETs, where the segmentation makes the classification task interpretable and extra cues from the dilated pancreatic duct are incorporated in proposed framework; 3) our overall framework achieves a sensitivity of 89.47% at a specificity of 81.08%, which indicates the potential direction to a clinical impact.

5.2 Method

5.2.1 The Overall Framework

We denote the dataset as $\mathbf{D} = \{(\mathbf{X}_1, y_1), \ldots, (\mathbf{X}_N, y_N)\}$, where N is total number of CT cases, $\mathbf{X}_n \in \mathbb{R}^{W_n \times H_n \times L_n}$ is a 3D volume with each voxel defined as the Hounsfield Unit (HU), and $y_n \in \{0, 1\}$ is the case label, by which 0 means a normal case while 1 for an abnormal case. By abnormal/tumor we mean cases diagnosed with PNETs throughout the whole paper. Our goal is to design a model $\mathbb{M} : y = f(\mathbf{X})$ mapping a CT image to its state of being abnormal or not.

Some previous work suggested to classify medical images by directly using deep neural networks [101], [102], however, we claim that a better strategy is to perform tumor segmentation together with the classification task. This makes the prediction **interpretable** for the classification results from segmentation cues, by which radiologists can take a further investigation of the suspicious abnormal regions. But for a deep neural network doing direct classification, it is hard for radiologists to further check which regions are suspicious while the adopted segmentation-for-classification (S4C) [20] sheds light on the abnormality detection, which is more plausible. In addition, this harnesses voxelwise annotations as fully supervision into the classification model, so that the entire network can be better optimized. Different from [20] which did S4C for PDAC only on venous phase, we incorporate the dilated pancreatic duct information on both arterial and venous phases for PNETs, which can further improve the sensitivity.

On our dataset, each training case is associated with a segmentation mask $\mathbf{M}_n$ of the same dimension as $\mathbf{X_n}$, among which $m_{n,i} \in \{0, 1, 2, 3\}$ denotes the annotated categories for the i-th voxel. More specifically, a background voxel is labelled as 0, and 1 means the voxel is inside the normal pancreas regions, and 2 denotes a voxel inside the tumor regions. We would like to emphasize that besides normal/abnormal pancreas

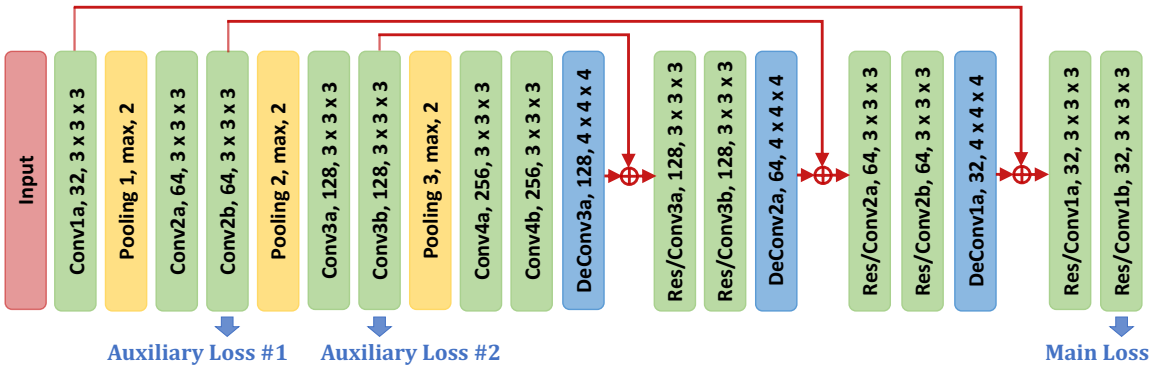

Figure 5.1. The network backbone of our S4C pipeline. We adopt an encoder-decoder fashion, where the encoder path on the left acts as a feature extractor to learn more and more compact features while the decoder path on the right decompresses the learned features gradually to obtain the dense predictions with higher and higher resolutions. The sum residual connections from the low-level layers are crucial to integrate the pixel-level features such as edges to the semantically meaningful features of high-level layers such as patterns or shapes. The two auxiliary losses serve as a deep supervision to reach a better optimization process, which favors the final segmentation performance [38]. The whole network is optimized with voxelwise softmax cross-entropy loss. The weight ratio for auxiliary losses #1, #2 and the main loss is $1 : 2 : 5$. Best viewed in color.

regions and background voxels, we annotate voxels inside dilated pancreatic duct regions as 3. This is motivated from the knowledge of radiologists that a pancreatic duct dilation is a sign of high risk for pancreatic cancer. If we predict a dilated pancreatic duct present for some cases where the PNETs are hardly invisible from textures, we can refine our judgment and would not miss those really hard cases. Note that a pancreas set includes the normal pancreas set, abnormal pancreas set and the dilated pancreatic duct set. The segmentation module is a mapping function $\mathbf{M} = \mathbf{s}(\mathbf{X})$, which is implemented by an encoder-decoder network mapping from CT scans with Hounsfield scale values to the categorical sets. The classification module is a binary function $y = c(\mathbf{M})$ with a set of rules given the segmentation as input that we will elaborate later. All in all, the whole framework is denoted as:

$$y = f(\mathbf{X}) = c \circ \mathbf{s}(\mathbf{X}). \tag{5.1}$$

5.2.2 Segmentation for Classification

Our segmentation backbone is shown in Fig. 5.1, which adopts the encoder and decoder [36] fashion for the dense prediction. The residual connections and auxiliary losses are the delicate designs aimed at a better and stable optimization [38]. The pooling layers of the encoder path compress the learning process into more compact feature space, from where the DeConv layers of the decoder path decompress them to semantically meaningful features in the fine-scale resolution. The whole framework takes the voxelwise softmax cross-entropy as the loss function, which shows stable and supreme performance on both normal pancreas and cystic pancreas segmentation [38]. The segmentation network takes patches as input, whose size is set to be 64×64×64, which covers sufficient context and makes memory for the networks design with powerful representation ability.

During training, we implemented simple yet effective augmentations on patches input, *i.e.*, rotation ($90°, 180°,$ and $270°$) and flip in all three axes (axial, sagittal and coronal), to increase the number of training samples which can alleviate the limited number of CT cases with annotations. During testing, we adopted the sliding window way to slide the whole CT volumes with a 20-voxel spatial stride. The overlapped regions are voted by majority. Based on the segmentation prediction, we classify each volume to be abnormal or normal. We compute the maximal connected component $\mathcal{C}_{\max}$ and keep a component which is either larger than 20% of $\mathcal{C}_{\max}$ or at a distance of less than 27 voxels to $\mathcal{C}_{\max}$. As for classification, a volume is predicted as PNETs if as least 40 voxels are predicted as tumors or 500 voxels are predicted as dilated pancreatic duct. To harness the dual-phase information, we classify a CT case as abnormal given any phase is predicted as PNETs, which improves the sensitivity at the cost of the specificity.

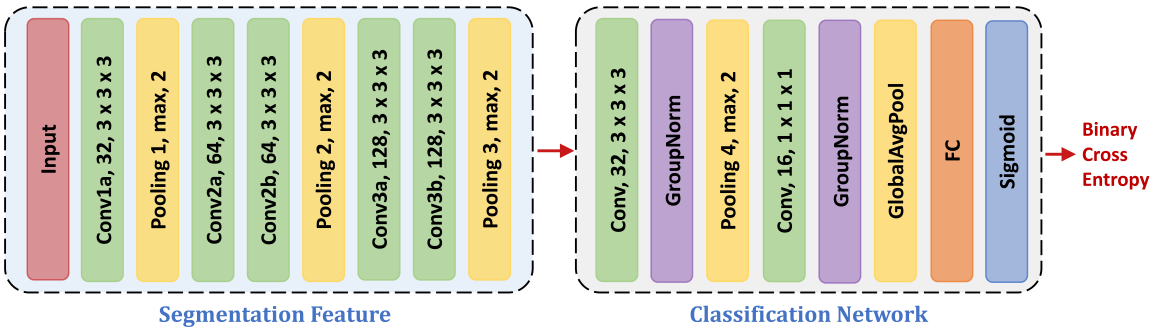

Figure 5.2. The classification network designed for the direct binary classification, *i.e.*, tumor versus non-tumor, as an ablation study. Best viewed in color.

5.2.3 Classification Network as Comparison

Since we adopt the Segmentation-for-Classification framework as our pipeline, it is natural to see how a classification network performs in comparison. Therefore, we implement a classification network as shown in Fig 5.2. To compare the S4C and classification network as fair as possible, we construct the classification network by feeding the features maps of segmentation network as input. This is due to the fact that the classification label is 0/1 per CT case, which owns much less information than the voxelwise 0/1's of segmentation labels. To filter out the large out-of-pancreas regions during training the classification network, an 128-way feature vector is extracted from the Region-of-Interest (RoI) of pancreas, which is derived from the ground-truth in the training or from the segmentation prediction with a margin in the testing. The 128-dimension Pool3 feature is chosen rather than the 256-dimension Conv4b feature vectors because of the better generalization ability we observed during our experiments. Since the feature map size is different for different size of pancreas RoIs, the batch size is chosen to be 1, then a GroupNorm [104] is added after each convolution in the classification to help the learning process. Note that the segmentation of pancreas is very good, which makes it doable for the RoI as input.

5.3 Experiments

5.3.1 Implementation Details

We collected a new dataset with 376 cases in total from potential renal donors, in which we have 148 normal cases and 228 biopsy-proven PNETs cases, where each case has both arterial and venous phases available. Four experts in abdominal anatomy voxelwisely annotated the pancreas, tumor regions, and dilated pancreatic duct using the Varian Velocity software, and checked by an experienced board-certified abdominal radiologist. For a radiologist expert, an average normal case took 20 minutes, and an average abnormal case 40 minutes to finish the voxelwise annotation. To quantitatively analyze our method, we adopt a same 4-fold cross-validation for S4C and classification network on these 376 cases in both phases. All in all, for a single phase, each training set contains 111 normal and 171 abnormal cases, and each corresponding testing set contains 57 abnormal and 37 normal cases. And the final quantitative performance is reported on the testing of all cases across 4 folds, by which we take every case into consideration to fully maximize the utilization of the medical data which are expensive and time-consuming to obtain. Our framework is implemented on Pytorch 0.5.0, and the GPU we are running on is the Tesla V100. The base learning rate is 0.01 and decayed polynomially (the power is 0.9) in a total of 80,000 iterations with a batch size of 16 for the S4C. The base learning rate is 0.001 and decayed polynomially (the power is 0.9) in a total of 20,000 iterations for the classification network. The weight decay and momentum are set to be 0.0005 and 0.9, separately. The total training time for a S4C model is 2.5 days while the average testing time for a case is around 10 mins while the training time for a classification is $\approx$ 12 mins given the segmentation features as input. All parameters are verified by the 4-fold cross-validation.

One of our goals is to quantify the segmentation accuracy by the Dice-Sørensen Coefficient (DSC) between the predicted and the ground-truth tumor regions $\mathcal{Y}$ and

$\mathcal{Y}^\star$, *i.e.*, $\text{DSC}(\mathcal{Y}, \mathcal{Y}^\star) = \frac{2\times|\mathcal{Y}\cap\mathcal{Y}^\star|}{|\mathcal{Y}|+|\mathcal{Y}^\star|}$. Our primary goal is to measure the abnormality classification by the sensitivity (the percentage of correctly classified abnormal cases) and the specificity (the percentage of correctly classified normal cases). In practice, there is always a trade-off between the sensitivity and specificity. We care much more about the sensitivity than the specificity since the final goal is to detect PNETs in the early stage for timely medical interventions.

5.3.2 Performance

From Table 5.1 that shows single-phase results, considering venous and arterial phases equally, our method in the venous phase achieves the best results on all evaluation matrix except for the comparable result with 3D UNet on the venous normal pancreas segmentation. On the one hand, the pancreas segmentation can be as high as 87.41% and 84.69% for the normal pancreas and abnormal pancreas respectively, which demonstrates the effectiveness of our method. On the other hand, the tumor segmentation performance is promising to be 43.11%, which outperforms the state-of-the-art segmentation networks, *i.e.*, 3D UNet [10] and VNet [11]. From the tumor segmentation results, it is a really challenging segmentation problem considering the various size, shape and locations of tumors. Note that a recent work on the PDAC segmentation achieves a DSC of $56.46 \pm 26.23\%$ [20], which is not as hard as the PNETs which are less common with even smaller voxel size. As for the abnormality classification task, our single phase model achieves 82.46% sensitivity at a specificity of 91.89%, which beats the second best 3D UNet by 0.88% and 2.70% respectively. To compare the same model in arterial and venous phases, we find that all three models behave generally better on the venous phase than the arterial phase. As in Fig. 5.3, our method performs better segmentation results for both venous and arterial phases, which shows the more powerful representation ability of our network backbone.

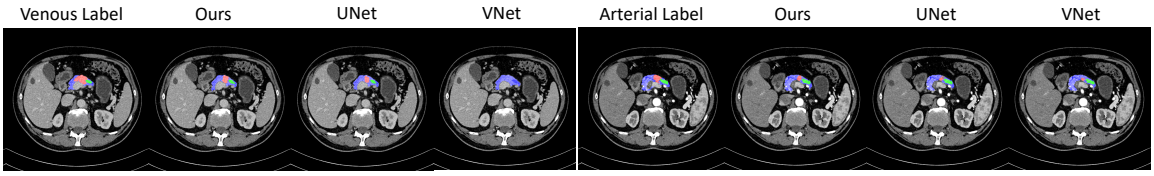

Figure 5.3. The segmentation visualization for the case number 7263. "Ours" method successfully detects the PNETs and dilated pancreatic duct regions on both the venous and the arterial phase, which performs better then "3D UNet" and "VNet".

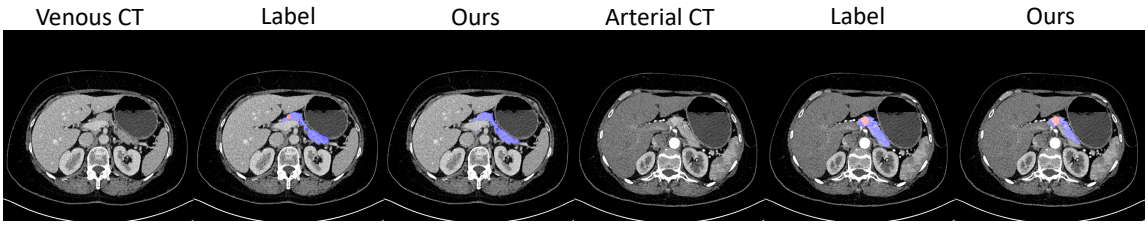

Figure 5.4. The segmentation visualization for the case number 7264. The tiny PNETs is hanging of the pancreas head, where "Ours" method successfully detects the PNETs regions on the arterial phase while missing the detection on the venous phase.

5.3.3 Dual-Phase Fusion and Comparison with Classification Network

In the clinical practice, the radiologists generally care much more about the sensitivity than they do about the specificity when it comes to the tumor detection. In radiology, some tissues are more visible in the venous phase while others are better to be captured in the arterial phase. Given we have CT scans in both arterial and venous phases available for each case, it is natural to think that we can combine the detection results from two phases together to take advantage of the different enhancement patterns when detecting the abnormality from different phases. We come up with a very straightforward way to combine the detection results. More specifically, if a model trained on any phase predicts this case to be abnormal, we regard this case to be abnormal. In this way, we are able to reduce the missing cases since a PNETs case can only be missed if both two phases classify the same case to be normal. The quantitative results are given in Table 5.2. First, our model beats both 3D UNet and

Phase	Method	N.Pan	A.Pan	Tumor	Misses	W.Calls	Sen	Spe
Venous	Ours	87.41%	**84.69%**	**43.11%**	**40/228**	**12/148**	**82.46%**	**91.89%**
Venous	UNet	**87.70%**	83.84%	41.43%	42/228	16/148	81.58%	89.19%
Venous	VNet	86.76%	83.90%	40.67%	49/228	20/148	78.51%	86.49%
Arterial	Ours	81.78%	83.34%	42.49%	44/228	21/148	80.70%	85.81%
Arterial	UNet	82.47%	83.33%	42.58%	44/228	20/148	80.70%	86.49%
Arterial	VNet	83.85%	82.79%	39.22%	43/228	31/148	81.14%	79.05%

Table 5.1. Performance of segmentation and classification on our own dataset in two different phases. From left to right: normal pancreas cases, abnormal pancreas cases and tumor segmentation accuracy (DSC, %), the number of missed abnormal cases out of 228 abnormal cases in total, the number of wrong calls of tumor predictions out of 148 normal cases in total, the corresponding sensitivity and the specificity.

VNet after the fusion as well. Second, in the trade-off by fuse two phases, we increase the sensitivity by 7.01% at the cost of the specificity drop by 10.81%, by which we value the fusion when it comes to the possible critical point of life or death for patients. As in Fig. 5.4, we visualize one case where our method misses the tumor prediction in the venous phase while detecting the tiny tumors successfully in the arterial phase.

From Table 5.2, S4C achieves the best in the sensitivity, which verifies the effectiveness of S4C framework. For the lower specific city of S4C than the classification network, we conjecture that the classification network is trained directly with a binary optimization goal and the feature map of segmentation as input can be favorable to classification network. However, the major drawback of the classification network is that it is notoriously hard to identify which regions in the original CT scans contribute to the final abnormality prediction. But, for our S4C framework, we provide radiologists with the predicted abnormal regions as a crucial cue for why we reach the decision. The convenience brought to radiologists for further diagnosis can be valued even with slightly lower specificity.

Phase	Method	Misses	W.Calls	Sensitivity	Specificity.
Arterial&Venous	S4C (Ours)	**24/228**	28/148	**89.47**%	81.08%
Arterial&Venous	3D UNet	26/228	31/148	88.60%	79.05%
Arterial&Venous	VNet	28/228	37/148	87.72%	75.00%
Arterial&Venous	Classification	24/228	**23/148**	89.47%	**84.46**%

Table 5.2. Performance of abnormality classification on our own dataset by considering two phases together. From left to right: the number of missed abnormal cases out of 228 abnormal cases in total, the number of wrong calls of tumor predictions out of 148 normal cases in total, the corresponding sensitivity and the specificity.

5.4 Conclusion and Future Works

In this work, we propose an overall framework to conduct the early detection of PNETs, the second common type of pancreatic cancer. We adopt the Segmentation-for-Classification strategy to make the classification result more interpretable to radiologists compared with a direct binary classification network. To quantitatively analyze our method, we voxelwisely annotate the largest PNETs CT dataset to the best of our knowledge. Furthermore, each CT case is collected in both arterial and venous phase, where the voxels of dilated pancreatic duct are annotated as well to increase the sensitivity in practice. Our approach outperforms the state-of-the-arts segmentation algorithms in terms of the DSC score and is comparable to a binary classification neural network in terms of sensitivity and specificity. In the future, we would like to integrate the classification network into the segmentation backbone, which can let these two tasks benefit from each other by a joint learning manner.

Chapter 6

V-NAS: Neural Architecture Search for Volumetric Medical Image Segmentation

Deep learning algorithms, in particular 2D and 3D fully convolutional neural networks (FCNs), have rapidly become the mainstream methodology for volumetric medical image segmentation. However, 2D convolutions cannot fully leverage the rich spatial information along the third axis, while 3D convolutions suffer from the demanding computation and high GPU memory consumption. In this paper, we propose to **automatically** search the network architecture tailoring to volumetric medical image segmentation problem. Concretely, we formulate the structure learning as **differentiable neural architecture search**, and let the network itself choose between 2D, 3D or Pseudo-3D (P3D) convolutions at each layer. We evaluate our method on 3 public datasets, *i.e.*, the NIH Pancreas dataset, the Lung and Pancreas dataset from the Medical Segmentation Decathlon (MSD) Challenge. Our method, named **V-NAS**, consistently outperforms other state-of-the-arts on the segmentation tasks of both normal organ (NIH Pancreas) and abnormal organs (MSD Lung tumors and MSD Pancreas tumors), which shows the power of chosen architecture. Moreover, the searched architecture on one dataset can be well generalized to other datasets, which demonstrates the robustness and practical use of our proposed method.

6.1 Introduction

Over the past few decades, medical imaging techniques, *e.g.*, magnetic resonance imaging (MRI), computed tomography (CT), have been widely used to improve the state of preventative and precision medicine. With the emerging of deep learning, great advancement has been made for medical image analysis in various applications, *e.g.*, image classification, object detection, segmentation and other tasks. Among these tasks, organ segmentation is the most common area of applying deep learning to medical imaging [14].

In this work, we focus on the volumetric medical image segmentation. Taking the pancreas and lung tumors segmentation from CT scans as an example as shown in Fig. 6.1, the main challenges lie in several aspects: 1) the small size of organs with respect to the whole volume; 2) the large variations in location, shape and appearance across different cases; 3) the abnormalities, *i.e.*, the pancreas and lung tumors, can change the texture of surrounding tissues a lot; 4) the anisotropic property along z-axis, which make the automatic segmentation even harder.

To tackle these challenges, many segmentation methods have been proposed in the literature. Starting from handcrafted features, there are methods proposed to use intensity thresholding [57], region growing [56], and deformable models [58], which often suffer from the limited feature representation ability and are less invariant to the large organ variations. With a huge influx of deep learning related methods, fully convolutional neural networks (FCNs), *e.g.*, 2D and 3D FCNs, have become the mainstream methodology in the segmentation area by delivering powerful representation ability and good invariant properties. The 2D FCNs based methods [18], [29], [30], [35], [36] perform the segmentation slice-by-slice from different views, then fuse 2D segmentation output to obtain a 3D result, which is a remedy against the ignorance of the rich spatial information. To make full use of the 3D context, 3D FCNs based

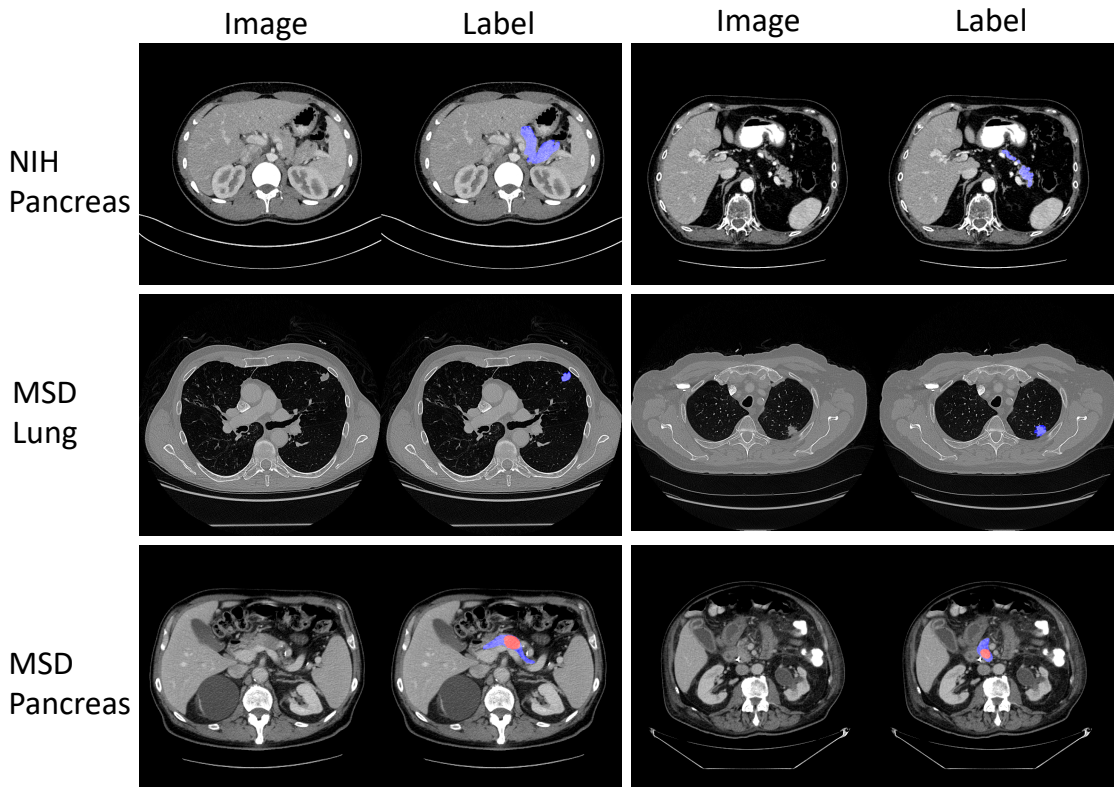

Figure 6.1. Typical examples from NIH Pancreas [18] in the 1st row, MSD Lung Tumors [19] in the 2nd row and MSD Pancreas Tumors [19] in the 3rd row. Two slices of different cases are randomly chosen from each dataset. Normal Pancreas regions are masked as blue and abnormal pancreas regions are masked as red. The lung cancers are masked as blue. Best viewed in color.

methods [10], [11], [37], [38] directly perform the volumetric prediction. However, the demanding computation and high GPU consumption of 3D convolutions limit the depth of neural networks and input volume size, which impedes the massive application of 3D convolutions. Recently, the Pseudo-3D (P3D) [39] was introduced to replace 3D convolution $k\times k\times k$ with two convolutions, *i.e.*, $k\times k\times 1$ followed by $1\times 1\times k$, which can reduce the number of parameters and show good learning ability in [40], [41] on anisotropic medical images. However, all the aforementioned existing works choose the network structure empirically, which often impose explicit constraints, *i.e.*, either 2D, 3D or P3D convolutions only, or 2D and 3D convolutions are separate from each other. These hand-designed segmentation networks with architecture constraints might not

be the optimal solution considering either the ignorance of the rich spatial information for 2D or the demanding computations for 3D.

Drawing inspiration from recent success of Neural Architecture Search (NAS), we take one step further to let the segmentation network **automatically** choose between 2D, 3D, or P3D convolutions at each layer by formulating the structure learning as **differentiable neural architecture search** [42], [43]. To the best of our knowledge, we are one of the first to explore the idea of NAS/AutoML in medical imaging field. Previous work [44] used reinforcement learning and the search restricts to 2D based methods, whereas we use differentiable NAS and search between 2D, 3D and P3D, which is more effective and efficient. Without pretraining, our searched architecture, named V-NAS, outperforms other state-of-the-arts on segmentation of normal pancreas, the abnormal lung tumors and pancreatic tumors. In addition, the searched architecture on one dataset can be well generalized to others, which shows the robustness and potential clinical use of our approach.

6.2 Related Work

6.2.1 Medical Image Segmentation

The volumetric medical image segmentation has been dominated by deep convolutional neural networks based methods in recent years. [36] proposed the UNet architecture tailored to tackle medical image analysis problems in 2D, which is based on an encoder-decoder framework: the encoder is designed to learn higher and higher level representations while the decoder decompresses compact features into finer and finer resolution to obtain dense prediction. Then, a similar approach was presented by [10] to extend UNet to 3D input. Later on, VNet [11] proposed to incorporate residual blocks penalized by the Dice loss rather than the cross-entropy loss on 3D data, which directly minimizes the used segmentation error measurement. Meanwhile, a few recent

works have been proposed to combine 2D and 3D FCNs as a compromise to leverage the advantages of both sides. [64] adopted a 3D FCN by feeding the segmentation predictions of 2D FCNs as input together with 3D images. H-DenseUNet [65] hybridized a 2D DenseUNet for extracting intra-slice features and a 3D counterpart for aggregating inter-slice contexts. Similarly, 2D FCNs and 3D FCNs are not optimized at the same time in [64], [65].

6.2.2 Neural Architecture Search

Neural Architecture Search (NAS) is the process of automatically discovering better neural architectures than human designs. We summarize the progress in along two dimensions: search algorithm and dataset/task.

Many NAS algorithms belong to either reinforcement learning or evolutionary algorithm. In the reinforcement learning formulation [67], the actions generated by an agent define the network architecture, and the reward is the accuracy on the validation set. In the evolutionary formulation [68], architectures are mutated to produce better offsprings, again measured by validation accuracy. Although these algorithms are general, they are usually computationally costly. To address this problem, [69] progressively expand the search space in order to achieve better sample efficiency. Differentiable NAS approaches [42], [43], [70] utilize sharing among candidate architectures, and are arguably the most efficient family of algorithms to date.

At the same time, we also notice that the earlier papers [71]–[73] focused solely on MNIST or CIFAR10 dataset. Later, [67]–[69] searched for "transferable architectures" from the smaller CIFAR10 to the much larger ImageNet dataset. More recently, [74], [75] demonstrated the possibility to directly search for architectures on the ImageNet dataset. Finally, [42] extended NAS beyond image classification to semantic segmentation.

This paper sits at the frontier of both dimensions discussed above. We follow the

differentiable NAS formulation for its efficiency. In terms of application domain, we directly search on volumetric image segmentation data, which is more demanding and challenging than 2D image labeling.

6.3 Method

We define a **cell** to be a fully convolutional module, typically composed of several convolutional (Conv+BN+ReLU) layers, which is then repeated multiple times to construct the entire neural network. Our segmentation network follows the encoder-decoder [11], [36] structure while the architecture for each cell, *i.e.*, 2D, 3D, or P3D, is learned in a differentiable way [42], [43]. The whole network structure is illustrated in Fig. 6.2, where green Encoder and blue Decoder are in the search space. We start with depicting the detailed network structure in Sec. 6.3.1, and then describing the search space of green Encoder and blue Decoder in Sec. 6.3.2 and Sec. 6.3.3, respectively, followed by the optimization and search process in Sec. 6.3.4.

6.3.1 Basic Network Architecture

As shown in the upper part of Fig. 6.2, our task is to train a convolution neural network model to predict the voxel labels of a CT scan as input. Similar to the state-of-the-art segmentation networks U-Net [36], V-Net [11], 3D U-Net [10] and ResDSN [38], our overall network structure consists of a high-to-low resolution process as a feature extractor, and then recovers the resolution through a low-to-high process to obtain dense predictions. To downsample 3D feature maps from a high resolution to a low resolution, the "Conv-Max Pool Down" in the encoder path is implemented by a conv kernel of $1\times1\times1$ with a stride of $[2, 2, 1]$ followed by a MaxPool $1\times1\times2$ with a stride of $[1, 1, 2]$. The counterpart along the decoder path is realized by the "Up" module to upsample 3D feature maps from a low resolution to a high resolution. More specifically, the "Up" layer first projects the input feature map to match the number

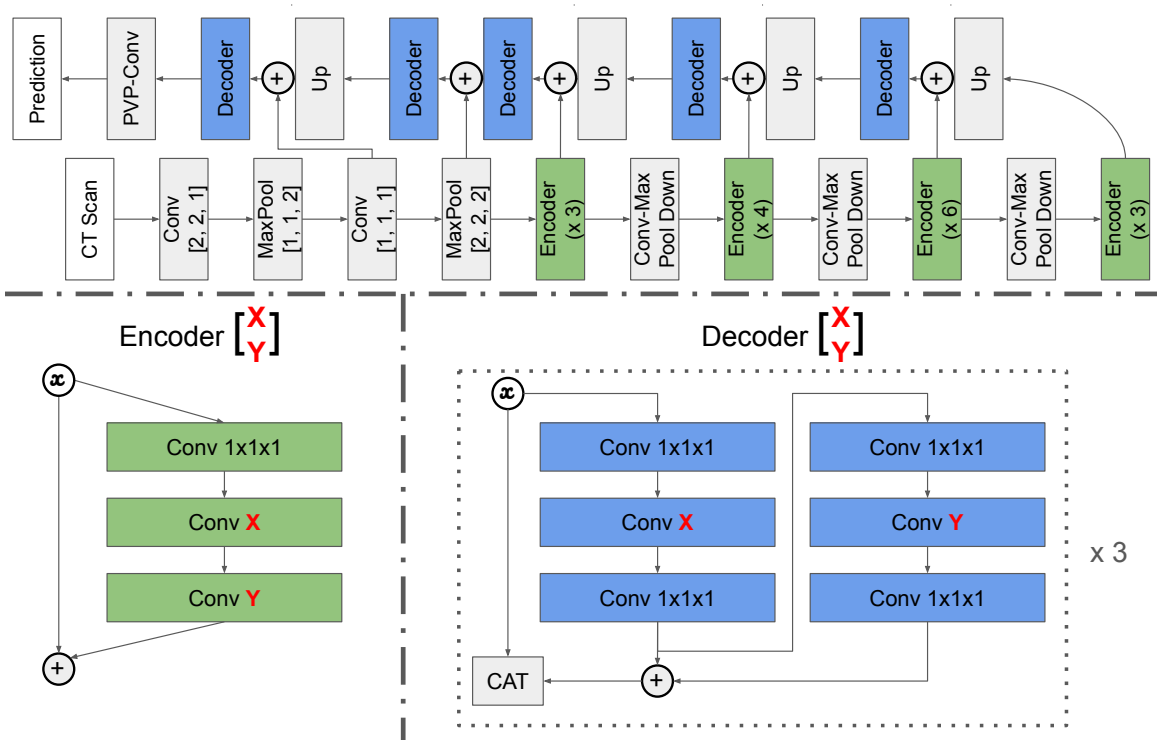

Figure 6.2. The segmentation network architecture. Each Encoder cell and Decoder cell has two candidate conv layers X and Y which are chosen between 2D, 3D, or P3D, whose details are defined in Sec. 6.3.2 and Sec. 6.3.3. The Encoder along the encoding path is repeated by $3, 4, 6, 3$ times while the decoder circled in the dashed rectangle is repeated by 3 times. The encoder path is designed from ResNet-50, while the decoder path takes advantage of dense block and pyramid volumetric pooling (PVP). The first two convolutional layers adopt a kernel size $7\times7\times1$ with stride $[2, 2, 1]$ and $1\times1\times3$ with stride $[1, 1, 1]$. The overall network architecture is effectively verified by [40] while we add the searching process for color blocks to choose between 2D, 3D, and P3D.

of feature channels of the higher Encoder feature by a $1\times1\times1$ conv, followed by the 3D tri-linear interpolation and element-wise sum with the Encoder feature at a higher resolution. The residual connections from the lower-level encoder to the higher-level decoder aggregate more detailed information to semantic meaningful feature maps to give more accurate dense predictions. A pyramid volumetric pooling module [105] is stacked at the end of the decoder path before the final output layer for fusing multiscale features.

6.3.2 Encoder Search Space

The set of possible Encoder architecture is denoted as $\mathcal{E}$, which includes the following 3 choices (*c.f.*, Fig.6.2 for Encoder $\begin{bmatrix}\mathrm{X}\\ \mathrm{Y}\end{bmatrix}$):

$$\{\underbrace{\text{Encoder}\begin{bmatrix}3\times3\times1\\1\times1\times1\end{bmatrix}}_{E_0\text{: 2D}},\underbrace{\text{Encoder}\begin{bmatrix}3\times3\times3\\1\times1\times1\end{bmatrix}}_{E_1\text{: 3D}},\underbrace{\text{Encoder}\begin{bmatrix}3\times3\times1\\1\times1\times3\end{bmatrix}}_{E_2\text{: P3D}}\} \tag{6.1}$$

As shown in Eq. 6.1, we define 3 Encoder cells, consisting of the 2D Encoder E_0, 3D Encoder E_1, and P3D Encoder E_2. 3×3×1 is considered as 2D kernel. The input of the l-th cell is denoted as x^l while the output as x^{l+1}, which is the input of the $(l+1)$-th cell. Conventionally, the encoder operation $O_e^l \in \mathcal{E}$ in the l-th cell is chosen from one of the 3 cells, *i.e.*, either E_0, E_1, or E_2. To make the search space continuous, we relax the categorical choice of a particular Encoder cell operation O_e^l as a softmax over all 3 Encoder convolution cells. By Eq. 6.2, the relaxed weight choice is parameterized by the encoder architecture parameter $\boldsymbol{\alpha}$, where α_i^l determines the probability of encoder E_i in the l-th cell,

$$\begin{aligned} x^{l+1} &= O_e^l(x^l) \approx \bar{O}_e^l(x^l) \\ \bar{O}_e^l(x^l) &= \sum\nolimits_{i=0}^{2} \frac{\exp(\alpha_i^l)}{\sum_{j=0}^{2}\exp(\alpha_j^l)} E_i(x^l), \end{aligned} \tag{6.2}$$

where $l = 1, \ldots, L$.

6.3.3 Decoder Search Space

Similarly, the set of possible Decoder architectures is denoted as $\mathcal{D}$, consisting of the following 3 choices (*c.f.*, Fig. 6.2 for Decoder $\begin{bmatrix}\mathrm{X}\\ \mathrm{Y}\end{bmatrix}$):

$$\{\underbrace{\text{Decoder}\begin{bmatrix}3\times3\times1\\3\times3\times1\end{bmatrix}}_{D_0\text{: 2D}},\underbrace{\text{Decoder}\begin{bmatrix}3\times3\times3\\3\times3\times3\end{bmatrix}}_{D_1\text{: 3D}},\underbrace{\text{Decoder}\begin{bmatrix}3\times3\times1\\1\times1\times3\end{bmatrix}}_{D_2\text{: P3D}}\} \tag{6.3}$$

As given in Eq. 6.3, we define 3 Decoder cells, composed of the 2D Decoder D_0, 3D Decoder D_1, and P3D Decoder D_2. The Decoder cell is defined as dense blocks, which shows powerful representation ability in [40], [65]. The input of the b-th Decoder cell is denoted as x^b while the output as x^{b+1}, which is the input of the $(b+1)$-th Decoder cell. The decoder operation O_d^b of the b-th block is chosen from either D_0, D_1, or D_2. As shown in Eq. 6.4, we also relax the categorical choice of a particular decoder operation O_d^b as a softmax over all 3 Decoder convolution cells, parameterized by the decoder architecture parameter $\boldsymbol{\beta}$, where β_i^b is the choice probability of decoder D_i in the b-th dense block,

$$\begin{aligned} x^{b+1} &= O_d^b(x^b) \approx \bar{O}_d^b(x^b) \\ \bar{O}_d^b(x^b) &= \sum_{i=0}^{2} \frac{\exp(\beta_i^b)}{\sum_{j=0}^{2} \exp(\beta_j^b)} D_i(x^b), \end{aligned} \tag{6.4}$$

where $b = 1, \ldots, B$.

6.3.4 Optimization

After relaxation, our goal is to jointly learn the architecture parameters $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$ and the network weights w by the mixed operations. The introduced relaxations in Eq. 6.2 and Eq. 6.4 make it possible to design a differentiable learning process optimized by the first-order approximation as in [43]. The algorithm for searching the network architecture parameters is given in Alg. 1. After obtaining optimal encoder and decoder operations O_e^l and O_d^b by discretizing the mixed relaxations $\bar{O}_e^l$ and $\bar{O}_d^b$ through `argmax`, we retrain the searched optimal network architectures on the $\mathcal{S}_{\text{trainval}} = \{\mathcal{S}_{\text{train}}, \mathcal{S}_{\text{val}}\}$ and then test it on $\mathcal{S}_{\text{test}}$.

Algorithm 1: V-NAS

Partition the whole labeled dataset $\mathcal{S}$ into the **disjoint** $\mathcal{S}_{\text{train}}$, $\mathcal{S}_{\text{val}}$ and $\mathcal{S}_{\text{test}}$;
Create the mixed operations $\bar{O}_e^l$ and $\bar{O}_d^b$ parametrized by α_i^l and β_i^b, respectively;
while *training not converged* **do**
1. Update weights w by descending $\nabla_w \mathcal{L}_{\text{train}}(w, \boldsymbol{\alpha}, \boldsymbol{\beta})$
2. Update $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ by descending $\nabla_{\boldsymbol{\alpha},\boldsymbol{\beta}} \mathcal{L}_{\text{val}}(w, \boldsymbol{\alpha}, \boldsymbol{\beta})$

Replace the relaxed operation $\bar{O}_e^l$ with $O_e^l = E_i, i = \texttt{argmax}_k \exp(\alpha_k^l)/\sum_{j=0}^{2} \exp(\alpha_j^l)$;
Replace the relaxed operation $\bar{O}_d^b$ with $O_d^b = D_i, i = \texttt{argmax}_k \exp(\beta_k^b)/\sum_{j=0}^{2} \exp(\beta_j^b)$;
Retrain the discretized architecture on the $\mathcal{S}_{\text{trainval}}$.

6.4 Experiments

6.4.1 Implementation Details

In this work, we consider a network architecture with L=3+4+6+3=16 and B=5, shown as color blocks in Fig. 6.2. The search space contains 3^{L+B}=3^{21}≈10^{10} different architectures, which is huge and challenging. The architecture search optimization is conducted for a total of 40,000 iterations. When learning network weights w, we adopt the SGD optimizer with a base learning rate of 0.05 with polynomial decay (the power is 0.9), a 0.9 momentum and weight decay of 0.0005. When learning the architecture parameters $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$, we use Adam optimizer with a learning rate of 0.0003 and weight decay 0.001. Instead of optimizing $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ from the beginning when weights w are not well-trained, we start updating them after 20 epochs. After the architecture search is done, we retrain weights w of the optimal architecture from scratch for a total of 40,000 iterations. The searching process takes around 1.2 V100 GPU days for one partition of train, val and test. All our models are trained on one V100 GPU with a customized batch size tuned to take full usage of the GPU memory due to different size input, which is computationally efficient in terms of neural architecture search task brought by the patch input.

In order to evaluate our method in the 4-fold cross-validation manner to fairly compare with existing works, we randomly divide a dataset into 4 folds, where each fold is evaluated once as the $\mathcal{S}_{\text{test}}$ while the remaining 3 folds as the $\mathcal{S}_{\text{train}}$ and $\mathcal{S}_{\text{val}}$ with a train *v.s.* val ratio as $2:1$. Therefore, there are in total 4 architecture search processes considering the 4 different $\{\mathcal{S}_{\text{train}}, \mathcal{S}_{\text{val}}\}$. The searched architecture might be different for each fold due to different $\{\mathcal{S}_{\text{train}}, \mathcal{S}_{\text{val}}\}$. In this situation, the ultimate architecture is obtained by summing the choice probabilities ($\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$) across the 4 search processes and then discretize the aggregated probabilities. Finally, we retrain the optimal architecture on each $\mathcal{S}_{\text{trainval}}$ and evaluate on the corresponding $\mathcal{S}_{\text{test}}$. All our implemented experiments use the same split of cross-validation and adopt Cross-Entropy loss, evaluated by the Dice-Sørensen Coefficient (DSC) formulated as $\text{DSC}(\mathcal{P}, \mathcal{Y}) = \frac{2\times|\mathcal{P}\cap\mathcal{Y}|}{|\mathcal{P}|+|\mathcal{Y}|}$, where $\mathcal{P}$ and $\mathcal{Y}$ denote for the prediction and ground-truth voxels set for a foreground class, respectively. This evaluation measurement ranges in $[0, 1]$ where 1 means a perfect prediction. We conduct experiments on 3 public datasets, *i.e.*, the NIH Pancreas dataset, the Pancreas and Lung dataset from the Medical Segmentation Decathlon (MSD) Challenge. And ablation studies are done on the NIH Pancreas dataset.

6.4.2 NIH Pancreas Dataset

We conduct experiments on the NIH pancreas segmentation dataset [18], which contains 82 normal abdominal CT volumes. The size of CT volumes is $512 \times 512 \times D$, where the number of slices D is different for different cases, ranging in $[181, 466]$. The physical spatial resolution for one voxel is $w \times h \times d$, where $d = 1.0$mm and $w = h$ that ranges from 0.5mm to 1.0mm. For the data pre-processing, we simply truncate the raw Hounsfield Unit (HU) values to be in $[-100, 240]$ and then normalize each raw CT case to have zero mean and unit variance to decrease the data variance caused by the physical processes [98] of medical images. As for the data augmentation

Method	Categorization	Mean DSC	Max DSC	Min DSC
V-NAS (Ours)	Search	$\mathbf{85.15 \pm 4.55}\%$	91.18%	**70.37**%
Baseline (Ours)	Mix	$84.36 \pm 5.25\%$	91.29%	67.20%
Xia *et al.* [64]	2D/3D	$84.63 \pm 5.07\%$	**91.57**%	61.58%
Zhu *et al.* [38]	3D	$84.59 \pm 4.86\%$	91.45%	69.62%
Yu *et al.* [106]	2D	$84.50 \pm 4.97\%$	91.02%	62.81%
Cai *et al.* [35]	2D	$82.40 \pm 6.70\%$	90.10%	60.00%
Zhou *et al.* [30]	2D	$82.37 \pm 5.68\%$	90.85%	62.43%
Dou *et al.* [37]	3D	$82.25 \pm 5.91\%$	90.32%	62.53%
Roth *et al.* [29]	2D	$78.01 \pm 8.20\%$	88.65%	34.11%
Roth *et al.* [18]	2D	$71.42 \pm 10.11\%$	86.29%	23.99%

Table 6.1. Comparison with other state-of-the-arts on the NIH Pancreas dataset evaluated by the 4-fold cross validation. Our one-stage segmentation network outperforms two-stage coarse-to-fine state-of-the-arts [38], [64]. The "Categorization" column categorizes each method by whether the segmentation method is based on 2D, 3D, or by the dynamic searching in our proposed method. The architecture searched on the NIH Pancreas dataset is coded as [0 0 0, 0 0 0 1, 2 0 2 0 2 2, 0 0 0] for the 16 Encoder cells, and [0 0 1 0 1] for the 5 Decoder blocks.

in the training phase, we adopt simple yet effective augmentations on all training patches, *i.e.*, rotation $(90°, 180°,$ and $270°)$ and flip in all three axes (axial, sagittal and coronal), to increase the number of 3D training examples which can alleviate the scarce of CT scans with expensive human annotations. Our training and testing procedure take patches as input to make more memory for the architecture design, where the training patch size is $96\times96\times64$ and the testing patch size is $64\times64\times64$ for the fine scale testing.

As shown in Table 6.1, our searched optimal architecture outperforms recent state-of-the-arts [38], [64], [106] segmentation algorithms. It is well worth noting that state-of-the-arts [38], [64] adopt a two-stage coarse-to-fine framework to have an extra segmentation network to refine the initial segmentation maps whereas our method outperforms them by only one stage segmentation, which is more efficient and effective. We also obtain the smallest standard deviation and the highest Min DSC, which demonstrates the robustness of our segmentation across all CT cases.

Furthermore, we implement the "Mix" baseline that equally initializes all architecture parameters $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ and keep them frozen during the training and testing procedures, which basically means the output takes exactly equal weight from 2D, 3D, and P3D in the encoder and decoder paths. Quantitatively, the search mechanism outperforms the "Mix" baseline by 3.17% and 0.79% in terms of the Min and Mean DSC, respectively, which verifies the effectiveness of the searching framework.

In details, we code the searched optimal architecture on the NIH Pancreas dataset by [0 0 0, 0 0 0 1, 2 0 2 0 2 2, 0 0 0] for the 16 Encoder cells, and [0 0 1 0 1] for the 5 Decoder blocks, where "0", "1" and "2" individually denote for the 2D, 3D and P3D, which are derived from definitions given in the Eq. 6.1 and Eq. 6.3. We observe that 2D convolutions are mostly picked up in the beginning for encoders while P3D appears in the intermediate encoders, and 3D convolutions are mostly chosen in the ending decoders. We hypothesize that 2D layer is efficient to extract the within-slice information coupled with the P3D to fuse learned feature maps in the intermediate stage while 3D kernels are effective in the semantic meaningful layers close to the output prediction.

We visualize two slices randomly chosen from three NIH pancreas cases as shown in Fig. 6.3. For the Case "#72" with a DSC of 90.96%, the pancreas appearance and boundary are well-captured and distinguished from its surroundings. For the Case "#81" with a DSC close to the "Mean DSC", the pancreas regions are generally predicted well though with some minor under-estimations near the head. As for the Case "#42" with the min DSC, the "VNAS" makes mistakes in the condition where the surrounding tissues are very complicate and the boundaries are ambiguous.

6.4.3 MSD Lung Tumors

We also evaluate our framework on the Lung Tumors dataset from the Medical Segmentation Decathlon Challenge (MSD) [19], which contains 64 training and 32

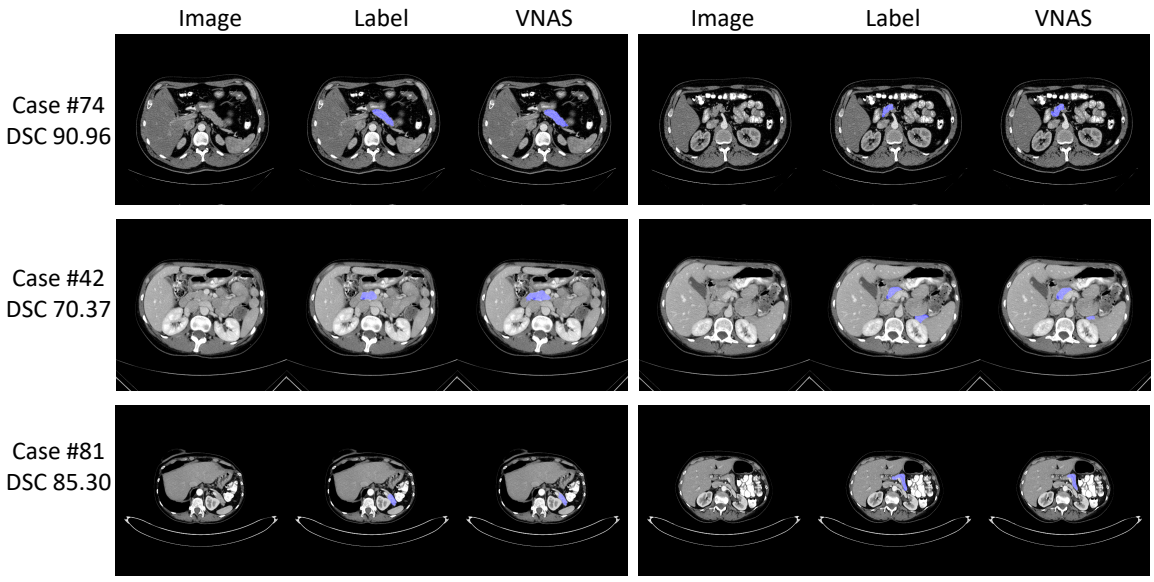

Figure 6.3. The visualization illustration of predicted segmentation for "VNAS" on the NIH Pancreas dataset. Two slices from Case "#74", "#42" and "#81" are randomly selected for visualization. The "Min DSC" Case "#42" and an average DSC Case "#81" are chosen. Blue masked regions denote for the pancreas voxels. Best viewed in color.

testing CT scans. It is aimed for the segmentation of a small target (lung tumors) in a large image, where only the lung cancers are labelled and to be segmented. Since the testing label is not available and the challenge panel is currently closed, we report and compare results of 4-fold cross-validation on the available 64 training set. The truncation range is set to be $[-1000, 1000]$ to cover almost all the lung HU values in the data pre-processing while the data augmentation is the same as mentioned in Sec. 6.4.2. More specifically, the patch size is set to be $64\times64\times64$ for the training and testing on MSD Lung Tumors dataset.

As given in Table 6.2, our method (V-NAS-Lung) beats 3D UNet [10] and VNet [11] by a large margin, at least 2.33% in terms of the "Mean DSC". The search process consistently outperforms the "Mix" version which takes equally the 2D, 3D and P3D as a fixed configuration. It is worth noting that the "Max DSC" of ours falls behind 3D UNet and VNet. We conjecture that since the overall network architecture is configured by the average choice probabilities of parameters $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ on 4 splits, our

Method	Categorization	Mean DSC	Max DSC	Median
V-NAS-Lung (Ours)	Search	$\mathbf{55.27 \pm 31.18}\%$	90.32%	66.95%
V-NAS-NIH (Ours)	Search	$54.01 \pm 31.39\%$	92.17%	**68.93**%
Baseline (Ours)	Mix	$52.27 \pm 31.40\%$	89.57%	61.71%
3D UNet	3D	$52.94 \pm 31.28\%$	93.58%	61.08%
VNet	3D	$50.47 \pm 31.37\%$	**93.85**%	57.82%

Table 6.2. Performance of different methods on the MSD Lung tumors dataset evaluated by the same 4-fold cross validation. The searched architecture on Lung tumors is coded as [0 0 0, 1 2 0 1, 2 1 2 0 0 0, 0 0 0] and [0 0 2 1 1]. It is worth noting that the searched architecture on the NIH dataset is well generalized to the Lung tumors dataset.

method tends to stably achieve the best overall segmentation performance, which is consistent with the much higher "Median DSC". More specifically, the searched architecture on Lung tumors is coded as [0 0 0, 1 2 0 1, 2 1 2 0 0 0, 0 0 0] and [0 0 2 1 1].

To take one step further, we report results of directly training the searched optimal architecture from the NIH Pancreas dataset (V-NAS-NIH) on the MSD Lung tumors dataset from scratch. The searched architecture generalizes well and achieves better performance than "Mix", 3D UNet and VNet. By comparing the two searched architectures from NIH Pancreas and MSD Lung Tumors datasets, we find that the two optimal architectures V-NAS-Lung and V-NAS-NIH share 68% (11 out of 16 Encoder cells) for the encoder path and 60% (3 out of 5 Decoder blocks) for the decoder path. The good property of transferring the network architecture searched on one dataset to another makes it possible for us to train the network architecture searched on a fairly big dataset with rich annotations to a small dataset with scarce annotations. We have not shown the "Min DSC" in the table since all approaches miss some lung tumors considering the lowest DSC to be 0, which shows that small lung tumors segmentation is a quite challenging task.

6.4.4 MSD Pancreas Tumors

Different from the NIH normal pancreas dataset, the MSD Pancreas Tumors dataset is labeled with both pancreatic tumors and normal pancreas regions. The original training set contains 282 portal venous phase CT cases, which are randomly split into 4 folds in our experiment, where each fold has its own training, validation and testing set and the final segmentation performance is reported on the average of 4 folds. Since the resolution along z-axis of this dataset is very low and number of slices can be as small as 37, the resolution of all cases on MSD Pancreas Tumors dataset are first re-sampled to an isotropic volume resolution of $d = 1.0$mm for each axis. Then the pre-processing and data augmentation is the same as Sec. 6.4.2. The patch size is set to be $64 \times 64 \times 64$ for both training and testing phases. Due to variant shapes and locations of tumors, the tumor segmentation is much more challenging and clinically important than the normal pancreas segmentation task since the early detection of pancreatic tumors can save lives.

As shown in Table 6.3, our searched architecture consistently outperforms 3D UNet and VNet, especially the pancreas tumors DSC delivers an improvement of at least 1.79%, which is regarded as a fairly good advantage. The 7.68% improvement over the manual "Mix" setting on the pancreas tumors consistently proves the advantage of the architecture search in the volumetric image segmentation domain. In details, the searched architecture on this dataset is coded as [0 2 2, 2 0 0 0, 2 2 1 2 1 1, 0 1 1] and [1 0 2 0 1], by which we observe there are more P3D and 3D convolutions selected compared with the searched optimal architecture from the NIH normal Pancreas dataset. We hypothesize that the between-slice information is very important to detect abnormalities since radiologists need to scroll up and down when reading CT scans to help the diagnosis.

We illustrate the visualization results of different methods as given in Fig. 6.4 on

Method	Categor.	Pancreas Tumors DSC			Pancreas DSC		
		Mean	Max	Median	Mean	Max	Min
V-NAS (Ours)	Search	**37.78 ± 32.12**%	92.49%	**38.32**%	**79.94 ± 8.85**%	**92.24**%	36.99%
Baseline (Ours)	Mix	30.10 ± 31.40%	92.95%	18.05%	78.41 ± 9.40%	92.21%	40.08%
3D UNet	3D	35.61 ± 32.20%	**93.66**%	32.23%	79.20 ± 9.43%	91.95%	**40.72**%
VNet	3D	35.99 ± 31.27%	92.95%	35.91%	79.01 ± 9.44%	92.05%	28.15%

Table 6.3. Performance of different methods on the MSD Pancreas tumors dataset evaluated by the same 4-fold cross validation. The results are given on the normal pancreas regions and pancreatic tumors, respectively. The searched architecture on Pancreas tumors dataset is coded as [0 2 2, 2 0 0 0, 2 2 1 2 1 1, 0 1 1] and [1 0 2 0 1].

the same slice of a same case for comparison in each row. 4 cases (#309, #021, #069 and #329) are chosen from the MSD Pancreas dataset, which are shown from top to bottom at each row, respectively. Note that the masked red and blue regions denote the pancreas tumor and normal pancreas regions, respectively. For the case #309 in the first row, the proposed "V-NAS" successfully detects the tiny tumor regions while "Mix" and "3D UNet" totally fails and "VNet" almost fails by only finding several tumor pixels. For the case #021, #069 and #329 from the 2nd to the 4th row, the searched architecture can semantically capture the tumor regions better because it can adaptively leverage both the rich 3D spatial context, the 2D within-slice information and the anisotropic structures.

6.4.5 Discussions

To further verify the advantage of automatically selecting among 2D, 3D and P3D convolution layers via the neural architecture search, we conduct ablation studies on manually choosing types in encoder and decoder paths to be purely either 2D, 3D or P3D on NIH Pancreas and MSD Lung Tumors datasets in this section.

6.4.5.1 Manual Setting on NIH Pancreas Dataset

As shown in Table 6.4, we manually configurate the architecture of Encoder and Decoder, where we train and test all configurations on the same 4-fold cross validation.

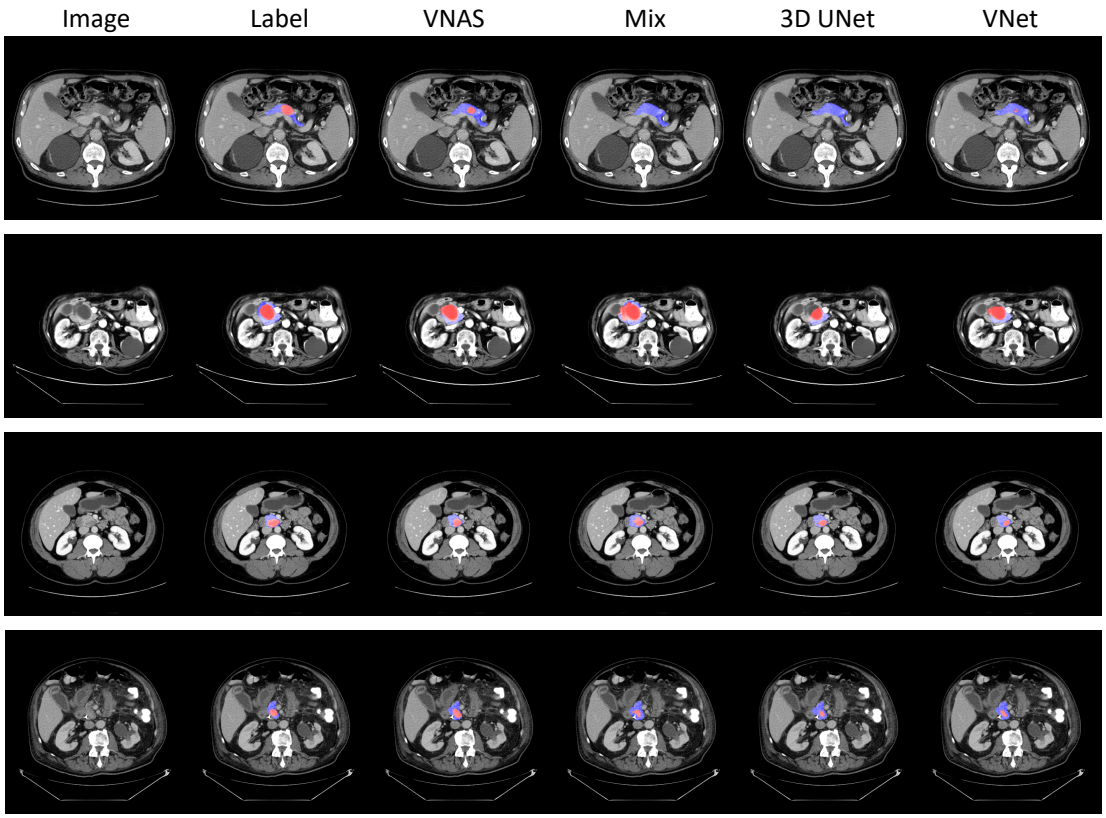

Figure 6.4. The visualization illustration of predicted segmentation for "VNAS", "Mix", "3D UNet" and "VNet" on the MSD Pancreas Tumors dataset, which is the most challenging task among our 3 segmentation tasks. Each row denotes a slice visualization from one case, and the specific cases numbers are "309", "021", "069" and "329" from top to bottom rows. The masked blue and red regions denote for the normal pancreas regions and tumor regions, respectively. Best viewed in color.

More specifically, all Encoders are set to be one type (2D, 3D, or P3D), and the same strategy is applied to the Decoders. Each row denotes the pure categorical choice for the Encoder cells while the column for the Decoder. We can find that 2D, 3D, and P3D kernels contribute differently to the segmentation by the experimental results. The P3D as Encoder and the P3D as Decoder achieve a mean DSC of 84.75% to outperform all other manual configurations. It is conjectured that the pure P3D takes most advantage of the anisotropic data annotation of the NIH dataset, where the annotation was done slice-by-slice along the z-axis. The different capability of learning semantic features between 2D, 3D and P3D for the dense volumetric image

Encoder\Decoder	3D	2D	P3D
3D	84.09%	83.77%	84.20%
2D	83.66%	83.29%	84.08%
P3D	84.32%	84.69%	**84.75%**

Table 6.4. Performance ("Mean DSC") of different encoder and decoder configurations on NIH dataset evaluated by the same 4-fold cross validation. The architecture is manually set with different choices from 2D, 3D and P3D. Ours obtains 85.15% in Table 6.1.

Method	Mean DSC	Max DSC	Median
3D/3D	$\mathbf{53.74 \pm 30.66}\%$	91.44%	60.55%
2D/2D	$52.01 \pm 31.50\%$	**92.58%**	63.27%
P3D/P3D	$51.48 \pm 32.46\%$	92.40%	**63.89%**

Table 6.5. Performance of different encoder and decoder configurations on MSD Lung Tumors evaluated by the same 4-fold cross validation. The architecture is manually configurated with different choices of 2D, 3D and P3D. Ours obtains 55.27% in Table 6.2.

segmentation problem drives us to naturally formulate it to be a neural architecture search task. As it turns out, the automatic selection among the 2D, 3D and P3D delivers the best performance with a mean DSC of 85.15% in Table 6.1.

6.4.5.2 Manual Setting on MSD Lung Tumors Dataset

On the MSD Lung Tumors dataset, we also report the manual architecture settings of 3D/3D, 2D/2D and P3D/P3D, *e.g.*, "3D/3D" stands for the configuration of only choosing 3D in both Encoder and Decoder cells. As given in Table 6.5, the "3D/3D" manual configuration achieves the best "Mean DSC" of $53.74 \pm 30.66\%$. We suspect that the lung cancers are located inside the lung organs, which needs the rich spatial context to predict the abnormality. Consistent with what we observe in Sec. 6.4.5.1, the neural architecture search idea outperforms all manual configurations to obtain a best mean DSC of $55.27 \pm 31.18\%$ with an advantage of 1.53% over the "3D/3D" in Table 6.2.

6.5 Conclusion and Future Works

We propose to integrate neural architecture search into volumetric segmentation networks to automatically find optimal network architectures between 2D, 3D, and Pseudo-3D convolutions. The search process is computationally efficient and effective. By searching in the relaxed continuous space, our method outperforms state-of-the-arts on both normal and abnormal organ segmentation tasks. Moreover, the searched architecture on one dataset can be well generalized to another one. In the future, we would like to expand the search space to hopefully find even better segmentation networks and reduce the computations.

Chapter 7

Detecting Scatteredly-Distributed, Small, and Critically Important Objects in 3D Oncology Imaging via Decision Stratification

Finding and identifying scatteredly-distributed, small, and critically important objects in 3D oncology images is very challenging. We focus on the detection and segmentation of oncology-significant (or suspicious cancer metastasized) lymph nodes (OSLNs), which has not been studied before as a computational task. Determining and delineating the spread of OSLNs is essential in defining the corresponding resection/irradiating regions for the downstream workflows of surgical resection and radiotherapy of various cancers. For patients who are treated with radiotherapy, this task is performed by experienced radiation oncologists that involves high-level reasoning on whether LNs are metastasized, which is subject to high inter-observer variations. In this work, we propose a divide-and-conquer decision stratification approach that divides OSLNs into tumor-proximal and tumor-distal categories. This is motivated by the observation that each category has its own different underlying distributions in appearance, size and other characteristics. Two separate detection-by-segmentation networks are trained per category and fused. To further reduce false positives (FP), we present a novel global-local network (GLNet) that combines high-level lesion characteristics with

features learned from localized 3D image patches. Our method is evaluated on a dataset of 141 esophageal cancer patients with PET and CT modalities (the largest to-date). Our results significantly improve the recall from 45% to 67% at 3 FPs per patient as compared to previous state-of-the-art methods. The highest achieved OSLN recall of 0.828 is clinically relevant and valuable.

7.1 Introduction

Measuring lymph node (LN) size and assessing its status are important clinical tasks, usually used to monitor cancer diagnosis and treatment responses and to identify treatment areas for radiotherapy. According to the Revised RECIST guideline [45], [46], only enlarged LNs with a short axis more than 10-15 mm in computed tomography (CT) images should be considered as abnormal. Such enlarged LNs have been the only focus, so far, of LN segmentation and detection works [7], [47]–[53]. However, in cancer treatment, besides the primary tumor, all metastasis-suspicious LNs are required to be treated. This includes the enlarged LNs, as well as smaller ones that are associated with a high positron emission tomography (PET) signal or any metastasis signs in CT. This larger category is regarded as oncology significant lymph nodes (OSLNs). Identifying the OSLNs and assessing their spatial relationship and causality with the primary tumor is a key requirement for a desirable cancer treatment outcome [54].

Identifying OSLNs can be a daunting and time-consuming task, even for experienced radiation oncologists. It requires using high-level sophisticated reasoning protocols and faces strong uncertainty and subjectivity with high inter-observer variability [25]. To the best of our knowledge, this problem has not been previously tackled in a fully automatized way. Our task on OSLNs detection is more challenging for the following reasons: (1) Finding OSLNs is often performed using radiotherapy CT (RTCT), which, unlike diagnostic CT, is not contrast-enhanced. (2) OSLNs exhibit low contrast with surrounding tissues and can be easily confused with other anatomical structures, *e.g.*,

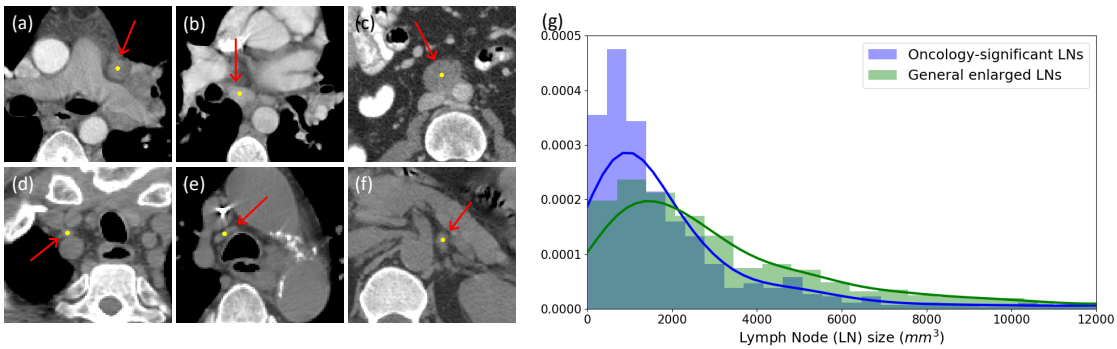

Figure 7.1. (a,b,c) Three examples of enlarged LNs, which all prior work targets, in *contrast-enhanced* CT. (d,e,f) Three instances of OSLNs, which our work focuses on, in non-contrast RTCT. This category has not been studied before as a computational task. (g) LN volume distributions for enlarged LNs from a public dataset [7], [53] and OSLNs in our radiotherapy dataset.

vessels and muscles, due to shape and appearance ambiguity. (3) The size and shape of OSLNs can vary considerably, and OSLNs are often scatteredly distributed at small size in a large spatial range of anatomy locations. See Fig. 8.1 for an illustration of the differences in appearance and size distribution between enlarged LNs the larger category of OSLNs. We can observe that OSLNs have higher frequencies at smaller sizes, challenging their detection. While, many previous works proposed automatic detection systems for enlarged LNs in contrast-enhanced CT [2], [7], [47], [48], [51], [53], [55], no work, as of yet, has focused on OSLN detection on non-contrast RTCT. Given the considerable differences between enlarged LNs and OSLNs, further innovation is required for robust and clinically useful OSLN detection.

Current clinical practices offer valuable insight in how to tackle this problem. For instance, physicians condition their analysis of suspicious areas based on their distance to the primary tumor. For LNs proximal to the tumor, physicians will more readily identify them as OSLNs for the radiotherapy treatment. However, for LNs far away from the tumor, physicians are more discriminating, only including them if there are clear signs of metastasis, such as enlarged in size, increased PET signals, and/or other CT-based evidence [107]. Hence, distance to the primary tumor plays a key role in

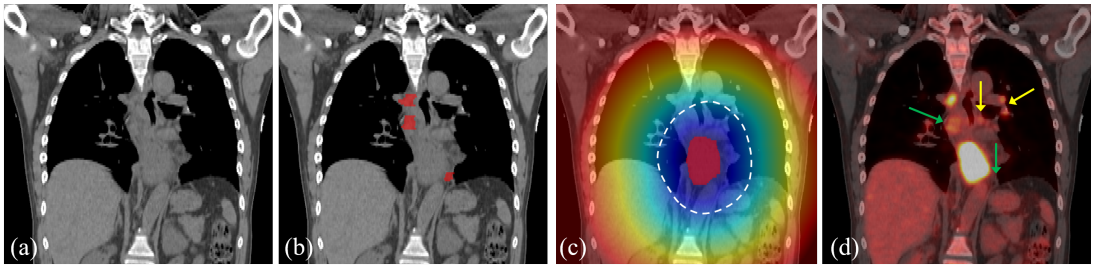

Figure 7.2. (a) A coronal view of RTCT for an esophageal cancer patient. (b) The manual annotated OSLN mask. (c) Tumor distance transform map overlaid on RTCT. The primary tumor is indicated by red mask in the center and the white dash line shows an example of the tumor proximal and distal region division. (d) PET imaging overlaid on RTCT. The yellow arrows show several FP PET signals, and the green arrows indicate two FN OSLNs where PET has weak or even no signals. A big central bright region in PET is the primary tumor region.

physician's decision making. Besides the distance, the PET modality is also highly important, as it significantly increases sensitivity [25]. However, PET is noisy and increased PET signals can often associate to normal physiological uptake. Moreover, PET only highlights $\sim 33\%$ of the OSLNs [108]. This is demonstrated in Fig. 7.2(d)'s example, where PET provides key information in identifying OSLNs, which might be too difficult to detect from RTCT only. Yet, the PET also exhibits false positives (FPs) and false negatives (FNs). Based on this observation, an effective method to leverage the complementary information in RTCT and PET is crucial, but this must be done with care.

To solve this problem, we emulate and disentangle the above practices. First, we propose and validate an intuitive and effective strategy that uses distance stratification to decouple the underlying OSLN distributions into two "tumor-proximal" and "tumor-distal" categories, followed by training separate networks to fit the class specific imaging features to the task. LNs that are spatially close to the primary tumor site are more suspicious (even if they are not enlarged); whereas spatially distal OSLNs may need to be identified with both CT and PET imaging evidence. This type of decision uncertainty stratification is evident in medical diagnosis and our work is one of the

first computational realizations. Second, for each OSLN category, we implement a 3D detection-by-segmentation framework that fuses predictions from two independent sub-networks, one trained on the RTCT imaging alone and the other learned via the early fusion (EF) of three channels of RTCT, PET and the 3D tumor distance map (Fig. 7.2(c)). RTCT depicts anatomical structures, which captures intensity appearance and contextual information, serves as a good baseline diagnostic imaging modality. In contrast, the EF stream takes into account PET's metastasis functional sensitivities as well as the tumor distance encoded in the distance transform map, which are both noisy but informative. Along with the distance stratification, this produces four predictions, which are all fused together as a late fusion (LF). This produces OSLN predictions that achieve sufficiently high sensitivities in finding OSLNs, which complements the high specificity but low sensitivity of human observers [25]. Missing true OSLNs can cause oncologically critical areas to remain untreated. Third, we propose a global-local network (GLNet) to further reduce the FP OSLN candidates obtained from above. The GLNet has two modules, with each module corresponding to the global or local spatial context. (1) For local context, we crop out any OSLN candidate region with certain context margins and adopt 3D residual convolutions [61], [109] to extract instance-wise localized deep feature maps. (2) For global context, we leverage the ontology-based medical knowledge from the large-scale NIH DeepLesion [2] dataset via a lesion tagging module [110], which provides high-level semantic information such as body part and shape/texture/size attributes that cannot be easily captured from local 3D image patches. The strategy of looking at locally (i.e., the imaging space) and globally (i.e., the semantic ontology space) is essential to mimic sophisticated clinical reasoning protocols. Both the imaging texture and appearance and semantically meaningful attributes are crucial to allow our workflow to filter out FPs while keeping sensitivities high. Our contributions can be summarized as follows:

- To the best of our knowledge, we are the first to address the clinically critical

task of detecting, identifying and characterizing OSLNs.

- We propose a novel 3D distance stratification strategy to divide and conquer the complex distribution of OSLNs into tumor-proximal and tumor-distal classes, to be solved separately, which emulates physician's decision process.
- Besides RTCT, we incorporate the PET imaging modality and 3D tumor distance maps into a two stream detection-by-segmentation network.
- We propose a novel GLNet to incorporate high-level ontology-derived semantic attributes of OSLNs with localized features computed from RTCT/PET.
- We collect and evaluate on the largest dataset to date on chest and abdominal radiotherapy. Our dataset comprises of 651 voxelwise-labeled OSLNs (by board-certified radiation oncologists) of 141 esophageal cancer patients. Our system significantly improves the detection recall from 45% to 67% at 3 FPs per scan, compared against the previous state-of-the-art CT-based detection method [78]. The highest achieved recall of 0.828 for OSLNs detection is also clinically relevant and valuable.

7.2 Related Work

Generic Lesion Detection: There are two popular approaches for generic lesion detection: end-to-end [76]–[79] and two-stage methods [80]–[83]. End-to-end methods have been extensively applied to the universal lesion detection task in the largest general lesion dataset currently available, *i.e.*, DeepLesion [2], and achieved encouraging performance. Notably, a multi-task universal lesion analysis network (MULAN) [78] so far achieves the best detection accuracy using a 3D feature fusion strategy and Mask R-CNN [84] architecture.

In contrast, two-stage methods explicitly divide the detection task into candidate

generation and FP reduction steps. The first step generates the initial candidates at a high recall and FP rate and the second step focuses on reducing the FP rate (especially the difficult ones) while maintaining a sufficient high recall. It decouples the task into easier sub-tasks and allows for the optimal design of each sub-task, which has shown to be more effective in problems like lung nodule [80], [83] and brain lacune [81] detection as compared to the one-stage method. We adopt the two-stage strategy for the OSLN detection to effectively incorporate different features, *i.e.*, PET imaging, tumor distance map and high-semantic lesion attributes, into each stage. We demonstrate the necessity of our strategy by comparing with the state-of-the-art (SOTA) universal lesion detector MULAN [78] in the experiment.

Lymph Node Detection and Segmentation: All previous works focus only on enlarged LN detection and segmentation in contrast-enhanced CT. Conventional statistical learning approaches [48]–[50], [55] employ hand-crafted image features, such as shape, spatial priors, Haar filters, and volumetric directional difference filters, to capture LN appearance and location. More recent deep learning methods achieve better performance. [47], [51], [52] applies the FCN or Mask R-CNN to directly segment LNs. In contrast, [7], [53] proposed a 2.5D patch-based convolutional neural network (CNN) with random view aggregation to classify LNs given all LN candidates already detected, and achieves SOTA classification accuracy for enlarged LNs. We demonstrate the effectiveness of the local and global modules in our GLNet compared with the 2.5D classification method [7].

Multi-Modal Image Analysis: The multi-modal imaging setup [111], [112] is a common and effective representation for segmenting anatomical structures in medical images. The pixel contrast and visual information in each modality is different and complementary for many applications. In our work, RTCT and PET have fundamentally different imaging physics, with RTCT corresponding to anatomy-based structural imaging and PET to functional imaging. Recent deep learning

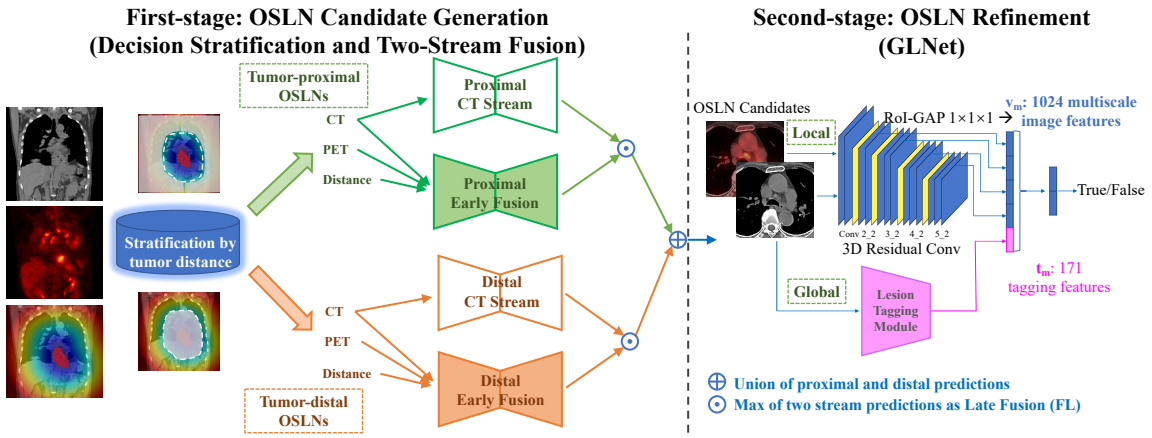

Figure 7.3. The overall framework of our 2-stage OSLN detection method. The 1st-stage adopts a divide-and-conquer distance stratification to divide OSLNs into tumor-proximal (green) and tumor-distal (orange) categories. For each category, a two-stream network, *i.e.*, CT stream (no fill) and CT, PET and tumor-distance early fusion stream (solid fill), is designed to learn the specific features for this category. After that, the predictions of the two streams are fused together via the "max" operation to achieve high recall. The GLNet of the 2nd-stage takes the OSLN candidates from the 1st-stage, and passes it through the local and global modules to reject FPs, leading to a final set of OSLNs with clinically relevant recall and low FPs.

approaches [83], [113]–[115] have exploited different fusion strategies for PET/CT, *e.g.*, early, late or chained fusion. In our 1st-stage, we propose a 2-stream deep network segmentation workflow (encoding RTCT alone or combined RTCT/PET and tumor distance map, respectively) and implement a concise late probability fusion scheme. This simple two-stream fusion strategy effectively generates the OSLN candidates with a high recall at a reasonable FP rate, which is desirable for the downstream 2nd-stage FP reduction.

7.3 Method

Fig. 8.2 illustrates our two-stage framework, which combines OSLN candidate generation with FP rejection. In the 1st-stage, we group OSLNs into two categories based on their distances to the primary tumor via distance stratification. For each category, a two-stream detection-by-segmentation network is designed to effectively incorporate

and fuse the RTCT and PET images, along with a tumor distance transform map. Results from two categories are merged together to produce the OSLN candidates. The goal of the 1st-stage is to have a set of OSLN candidates with high recall while keeping FPs to a reasonable number. In the 2nd-stage, the GLNet, composed of local and global modules, is proposed to serve as a selective classifier to reject OSLN FP candidates (especially the difficult ones) while preserving sufficient recall.

7.3.1 1st-Stage: Candidate Generation

Assuming N data samples, we denote a dataset as $\mathbf{S} = \left\{\left(\mathbf{X}_n^{\mathrm{CT}}, \mathbf{X}_n^{\mathrm{PET}}, \mathbf{Y}_n^{\mathrm{T}}, \mathbf{Y}_n^{\mathrm{LN}}\right)\right\}_{n=1}^{N}$, where $\mathbf{X}_n^{\mathrm{CT}}$, $\mathbf{X}_n^{\mathrm{PET}}$, $\mathbf{Y}_n^{\mathrm{T}}$ and $\mathbf{Y}_n^{\mathrm{LN}}$ represent the non-contrast RTCT, registered PET, the tumor mask and the ground truth LN segmentation mask, respectively. Without loss of generality we drop n for conciseness for the rest of the paper. The mask $\mathbf{Y}^{\mathrm{T}}$ is a 3D volume with a binary value y_i at each spatial location i to indicate whether the voxel x_i is the OSLN target. To encode the tumor distance information, we compute the 3D signed distance transform map from the primary tumor $\mathbf{Y}^{\mathrm{T}}$, denoted as $\mathbf{X}^{\mathrm{D}}$, where each voxel $x_i^{\mathrm{D}} \in \mathbf{X}^{\mathrm{D}}$ represents the distance between this voxel to the nearest boundary of the primary tumor. Let $\Gamma(\mathbf{Y}^{\mathrm{T}})$ be a function that computes boundary voxels of the tumor. The distance transform value at a voxel x_i^{D} is computed as

$$\mathbf{X}^{\mathrm{D}}(x_i^{\mathrm{D}}) = \begin{cases} \min\limits_{q \in \Gamma(\mathbf{Y}^{\mathrm{T}})} d(x_i^{\mathrm{D}}, q) & \text{if} \quad x_i^{\mathrm{D}} \notin \mathbf{Y}^{\mathrm{T}} \\ -\min\limits_{q \in \Gamma(\mathbf{Y}^{\mathrm{T}})} d(x_i^{\mathrm{D}}, q) & \text{if} \quad x_i^{\mathrm{D}} \in \mathbf{Y}^{\mathrm{T}} \end{cases}, \tag{7.1}$$

where $d(x_i^{\mathrm{D}}, q)$ is a distance measure from x_i^{D} to q. We choose to use Euclidean distance in our work and use Maurer's efficient algorithm [116] to compute the $\mathbf{X}^{\mathrm{D}}$. Note that $\mathbf{X}^{\mathrm{CT}}$ and $\mathbf{X}^{\mathrm{PET}}$ and $\mathbf{Y}^{\mathrm{T}}$ are already given and $\mathbf{X}^{\mathrm{D}}$ is pre-computed at the inference time.

We denote segmentation models as a mapping: $\mathbf{P} = \mathbf{f}(\mathcal{X}; \mathbf{\Theta})$, where $\mathcal{X}$ is a set of inputs, which may consist of a single modality or a concatenation of multiple modalities.

$\boldsymbol{\Theta}$ indicates model parameters, and $\mathbf{P}$ denotes the predicted probability volume. Specifically in a neural network, $\boldsymbol{\Theta}$ is parameterized by the network parameters.

7.3.1.1 Distance-Based Stratification

Based on $\mathbf{X}^{\mathrm{D}}$, we divide image voxels into two groups, x_{prox} and x_{dis}, to be tumor-proximal and tumor-distal, respectively, where $\mathrm{prox} = \{i | x_i^{\mathrm{D}} \leq d\}$ and $\mathrm{dis} = \{i | x_i^{\mathrm{D}} > d\}$. In this way, we divide all OSLNs into two categories, and train separate segmentation models for each. By doing this, we break down the challenging OSLN segmentation problem into two simpler sub-problems, each of which can be more easily conquered. This allows the OSLN segmentation method to emulate the clinician decision process, where tumor-proximal LNs are more readily considered oncology-significant, whereas a more conservative process, with differing criteria, is used for tumor-distal LNs. See Fig. 8.2 for the distance stratification demonstration. Prediction volumes generated by the tumor-proximal or tumor-distal models are denoted as $\mathbf{P}_{\mathrm{prox}}$ and $\mathbf{P}_{\mathrm{dis}}$, respectively.

7.3.1.2 Two-Stream Detection-by-Segmentation Fusion

For each OSLN category, we again emulate the physician's diagnostic process by fully exploiting the complementary information within the RTCT, PET and tumor distance map. Specifically, *for each OSLN category*, we design a two-stream 3D segmentation workflow that fuses predictions from two independent sub-networks, one trained using the RTCT alone (*CT stream*), and the other trained using the three channels of RTCT, PET and the tumor distance map jointly (*early fusion stream*). In this way we generate predictions based on only structural appearance, complementing them with additional predictions incorporating PET's auxiliary functional sensitivity and the tumor distance-map's location context. We denote prediction volumes from the RTCT and early fusion stream models as $\mathbf{P}_{(.)}^{\mathrm{CT}}$ and $\mathbf{P}_{(.)}^{\mathrm{EF}}$, respectively, where the subscript may be either "prox" or "dis" for the tumor-proximal or tumor-distal categories,

respectively. The result is four separate predictions. To ensure a high recall of OSLN detection in this stage, we apply a straightforward yet effective *late fusion* by taking the element-wise `max` and `union` operations of the four predictions:

$$\mathbf{P^{LF}} = \{p_i | p_i = \texttt{union}\{\texttt{max}\{p^{\text{CT}}_{\text{prox},i}, p^{\text{EF}}_{\text{prox},i}\}, \texttt{max}\{p^{\text{CT}}_{\text{dis},i}, p^{\text{EF}}_{\text{dis},i}\}\}\}, \tag{7.2}$$

where $p^{(.)}_{(.),i} \in \mathbf{P}^{(.)}_{(.)}$ and i indexes individual voxel locations. Stratifying OSLNs by tumor distance and performing two stream fusion are both crucial for a high recall.

From the final segmentation probability $\mathbf{P^{LF}}$, we derive the binary segmentation mask $\mathbf{B}$ by thresholding, and then calculate the OSLN instance candidates as the input to the 2nd-stage.

7.3.2 2nd-Stage: False Positive Reduction

The goal of the 2nd-stage is to reject as many FPs as possible while maintaining a sufficiently high recall. We first aggregate all predicted OSLN instances from the 1st-stage to be $\mathbf{R} = \left\{\left(\mathbf{C}^{\text{CT}}_m, \mathbf{C}^{\text{PET}}_m, l_m\right)\right\}^M_{m=1}$ as the OSLN candidates set, where $\mathbf{C}^{\text{CT}}_m$ and $\mathbf{C}^{\text{PET}}_m$ denote the local RTCT and PET image patches cropped at the mth OSLN candidate, respectively, and the binary scalar l_m is the label indicating if this instance is a true OSLN. We formulate a classification model: $q = \mathbf{g}(\mathbf{C}; \mathbf{\Phi})$, where $\mathbf{C}$ represents the input image patches, $\mathbf{\Phi}$ stands for model parameters, and q denotes the predicted probability. Here, when appropriate, we drop the m for simplicity.

To design a highly effective OSLN classifier, especially for the hard FPs, we propose a global and local network (GLNet) to leverage both local (CT appearance and PET signals) and global (spatial prior and other attributes) features. We describe their details in the following subsections.

7.3.2.1 Local Module in Global-Local Network

For the local module, we adopt a multi-scale 3D CNN model with a 3D ROI-GAP pooling layer [109] to extract OSLN local features from the image patch $\mathbf{C}$. Unlike

the 2.5D input patch used in [7], the 3D CNN explicitly uses 3D spatial information, improving classification performance. Either CT or CT+PET patches can be fed into the local model, and we evaluate both options. The features generated by each convolutional block separately pass through a 3D ROI-GAP pooling layer and a fully connected layer to form a 256D vector, which are then concatenated together to a multi-scale local representation for the OSLN instance. Since we use four CNN blocks, this leads to a total of $4 \times 256 = 1024$-dimensional feature vector, which is denoted as $\mathbf{v}$. See the 2nd-stage illustration in Fig. 8.2.

7.3.2.2 Global Module in Global-Local Network

For the global module, we migrate the ontology-based medical knowledge from the large-scale DeepLesion [2] dataset, via a pretrained lesion tagging module, *i.e.*, LesaNet [110]. Trained from radiology reports, LesaNet predicts high-level semantic lesion properties in the form of a 171-dimensional vector describing the lesion's body part, type and attributes. These information may not be easily captured from local image patches. We use the prediction of LesaNet on the mth OSLN candidates to generate a 171-dimensional feature vector $\mathbf{t}_m$, which provides complementary information to distinguish a true OSLN from false ones. For example, one body-part attribute from the feature vector indicates whether the lesion is in the "muscle", which may be confused with OSLNs when only analyzing the small local image patch, but are easier to identify under a global context. These kinds of FP candidates can be safely rejected using the global properties. LesaNet also predicts body parts like hilum LN, subcarinal LN, pretracheal LN and attributes like hypo-attenuation, tiny, oval, which are all relevant properties to distinguish true OSLNs from false ones.

To combine the strength of local image-based features and global OSLN properties, the GLNet concatenates $\mathbf{v}_m$ and $\mathbf{t}_m$ and passes through a fully connected layer to generate the final OSLN classification score, as illustrated in Fig. 8.2.

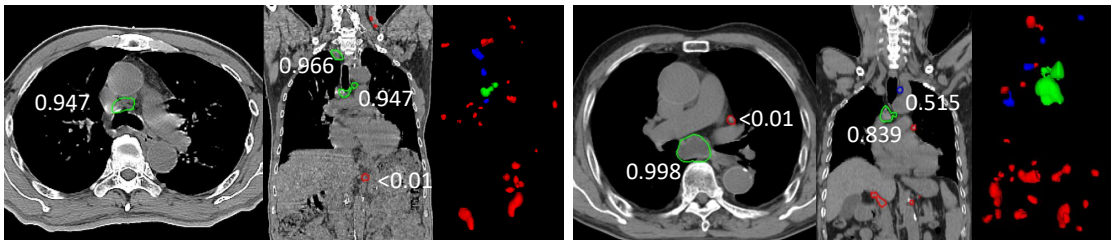

Figure 7.4. Visualization of segmentation contours in axial view or coronal views, and 3D mask volume rendering of two cases (left, and right). All masks/contours are LNs candidates from the first stage, where red ones are rejected in the 2nd-stage. Compared with ground truth LNs, TP and FP are colored in green and blue, respectively. Best viewed in color.

7.4 Experiments

In Fig. 7.4, we provide visual examples of our OSLN detection results. First, we can find that a large number of OSLN candidates are generated after the 1st-stage, to warrant a high recall. Second, the majority of OSLN candidates are effectively reduced by our proposed GLNet classifier, while the true positives in the 1st-stage are kept after the false positive reduction, which is desirable. Below we elaborate further on our experiments, providing dataset and implementation details along with extensive quantitative analyses.

7.4.1 Datasets

We collected an in-house dataset to evaluate our 1st-stage performance as well as the overall two-stage performance. We collected 141 non-contrast RTCTs of anonymized esophageal cancer patients, all undergoing radiotherapy treatments. Radiation oncologists labeled the 3D segmentation masks of the primary tumor and all OSLNs treated by radiotherapy. In total, there is a non-contrast RTCT scan and a PET/CT for each of the 141 patients and 651 OSLNs with voxel-wise labels in the mediastinum or upper abdomen regions. This is *the largest annotated OSLN dataset* in the chest and

abdominal region to-date. We register the PET images to RTCT using the registration method in [113]. For evaluation, we randomly split the annotated 141 patients into 84 for training, 23 for validation, and 34 for testing. In our experiments, we resample RTCT and PET images to have a consistent spatial resolution of $1 \times 1 \times 2.5$ mm. For data preprocessing, we truncate Hounsfield unit values of the RTCT to be within $[-200, 300]$. We also calculate the mean and standard deviation values of PET images across the entire training set and then normalize all PET images with these values.

7.4.2 Implementation Details

In the first stage, for training the OSLN detection-by-segmentation network, we crop sub-volumes of $96 \times 96 \times 64$ from the 3D images of RTCT, registered PET and the tumor-LN distance map. For the distance stratification, we set $d = 70$ mm to divide OSLN instances to tumor-proximal and tumor-distal sub-groups as suggested by our physician, and train the tumor-proximal and tumor-distal models separately. For data augmentation, we use straightforward and effective augmentations on training patches, *i.e.*, rotation (90°, 180°, and 270°) with a probability of 0.5 and flips in the axial view with a probability of 0.25. We can choose any popular segmentation network as our 1st-stage backbone, and we opt for the standard 3D UNet [10] as it gives the best performance in our network backbone ablation study in Sec. 7.4.4. Models are trained on two NVIDIA Quadro RTX 6000 GPUs with a batch size of 8 for 50 epochs. The RAdam [117] optimizer with a learning rate of 0.0001 is used with a momentum of 0.9 and a weight decay of 0.0005. For testing, we use a computationally efficient way to inference, *i.e.*, sub-volumes of $224 \times 224 \times 64$ are cropped along the vertical axis with the horizontal center the same as the center of lung masks [118]. These sub-volume predictions are aggregated to obtain the final OSLN segmentation results.

In the 2nd-stage, to train the local module of GLNet, the input images are generated by cropping a $48 \times 48 \times 32$ sub-volume centered around each predicted OSLN candidate

from the 1st-stage. If the size of the predicted OSLN is larger than $48 \times 48 \times 32$, we resize the sub-volume so that it contains at least an 8-voxel margin of the background along each dimension to ensure sufficient background context. The bounding boxes (bbox) for the 3D ROI-GAP pooling layer in Sec. 7.3.2 are generated by randomly jittering the bbox around the predicted OSLN with a 3-voxel range in each dimension. For the global module of GLNet, we use the publicly available LesaNet [110] pre-trained on the DeepLesion dataset. The input of LesaNet is a 120×120 2D CT image patch around the OSLN candidate. The overall GLNet is trained using Adam [119] optimizer with a learning rate of 0.0001 and batch size of 32 for 10 epochs.

7.4.3 Evaluation Metrics

We first describe the hit, *i.e.*, the correct detection, criteria for OSLN detection when using the segmentation results. For an OSLN prediction from the 1st-stage, if it overlaps with any ground-truth OSLN, we treat it as a hit provided that its estimated radius is similar to the radius of the ground-truth OSLN. After confirming with our physician, a predicted radius must be within a factor of $[0.5, 1.5]$ to the ground-truth radius.

7.4.3.1 Recall and Precision

We assess the performance of the 1st-stage by reporting the recall at a range of desired precision points. Note that the goal of the 1st-stage is to achieve a high recall (even with quite a few FPs) so that the 2nd-stage has a high upper-bound recall to work with while it filters out FPs. We report the mean recall (mRecall) at a precision range of $[0.10, 0.20]$ to reflect the model performance. We also report the recall at a precision of 0.15, which is the operating point we choose to generate inputs for the 2nd-stage. This operating point was chosen after confirming with our radiation oncologist. Both the recall and precision are macro-averaged across patients.

Backbone	Recall@0.15		mRecall@0.10-0.20	
	CT	EF	CT	EF
3D-UNet	**0.736**	**0.732**	**0.762**	**0.722**
SE-UNet	0.686	0.705	0.693	0.705
HRNet	0.524	0.656	0.538	0.638
PSNN	0.709	0.574	0.714	0.592

Table 7.1. Ablation study of different backbones for the CT and early fusion streams.

Input	Recall@0.15		mRecall@0.10-0.20	
	w/	w/o	w/	w/o
LF	**0.828**	**0.786**	**0.817**	0.732
EF	0.788	0.732	0.760	0.722
CT	0.772	0.736	0.772	**0.762**

Table 7.2. 3D UNet performance with ("w/") and without("w/o") distance stratfication. All three settings, CT, EF, and LF, are tested.

7.4.3.2 FROC

To evaluate both the complete workflow (1st+2nd-stage), we compute the free response operating characteristic (FROC), which measures the recall against different numbers of FPs allowed per patient. We report the average recall (mFROC) at 2, 3, 4, 6 FPs per patient study. Besides the mFROC, we also report the best F1 score a model can achieve.

7.4.4 1st-Stage Ablation Study

7.4.4.1 Segmentation Network Backbone

We evaluated different segmentation backbones for the OSLN candidate generation, *i.e.*, standard UNet [10], UNet with squeeze-and-excitation (SE) block [120], HRNet [121], and PSNN [113]. As shown in Table 7.1, the standard 3D UNet [10] consistently outperforms other backbones. For PSNN [113], it probably has difficulty handling this challenging task (dealing with small objects) due to its simplistic "upsampling" decoders. For the HRNet [121], due to its memory-hungry computations, we can

only add the high resolution features after two pooling layers, which is undesired for segmenting OSLNs. The attention module from the SE block [120] does not help with this segmentation task either.

7.4.4.2 Distance Stratification and Two-Stream Network Fusion

We verify the effectiveness of the proposed distance stratification method under different settings. As shown in Table. 7.2, among all settings, *i.e.*, CT, early fusion (EF), and late fusion (LF), the distance stratification consistently improves recall@0.15 by $4\% - 5\%$. Similar improvements are seen for mRecall@0.1-0.2. These results strongly support our use of distance stratification, which is shown to be effective under different input settings.

Table 7.2 also reveals the importance of using and fusing different streams. As we can see, the CT stream and the EF stream achieve similar performance to each other, regardless of whether distance stratification is used or not. However, when the two streams are combined together using LF, marked improvements are observed. For example, the recall@0.15 gains 4%-5%, and the mRecall@0.1-0.2 shows similar improvements. These quantitative results validate the effectiveness of the proposed distance stratification and the two-stream network fusion.

7.4.5 2nd-stage Ablation Study

7.4.5.1 Necessity of the 2nd-stage

To gauge the impact of the 2nd-stage, we first directly evaluate the OSLN detection accuracy using the 1st-stage alone. Specifically, the detection score of each OSLN instance is determined by averaging the segmentation probability for every voxel within the segmentation mask. All "1st-stage only" results in Tab. 8.1 are marked by "#". Focusing first on the LF setting, when using the 1st-stage alone it provides 0.441 F1 and 0.478 mFROC. When adding a second-stage classifier only accepting CT

as input, the F1 scores and mFROC are improved to 0.513 and 0.576, respectively. Providing the PET image and global tags to the 2nd-stage classifier boosts performance even further to 0.552 and 0.645 for F1 scores and mFROC, respectively. These are clinically impactful gains. Finally, regardless of the 1st-stage setting (LF, EF, or CT), the 2nd-stage classifier provides clear improvement. This proves the versatility and strength of our workflow.

7.4.5.2 Role of Local and Global Modules in GLNet

To show the necessity of both the local and global GLNet modules, we also evaluated purely local and purely global 2nd-stage classification performance. As can be seen in Table 8.1, regardless of which 1st-stage setting is used, a purely local 2nd-stage (*e.g.* last 2nd and 3rd rows) outperforms a purely global 2nd-stage (*e.g.* last 4th row). This indicates that the high-level semantic features migrated from the general lesion tagging model, *i.e.*, LesaNet [110], are less effective than the local OSLN features extracted from CT or CT+PET. However, when combining the global tags with the local patches using the proposed GLNet, mFROC performance is increased from 0.594 to 0.645 (when using the LF 1st-stage setting). This demonstrates that both local and global features contribute to our ultimate performance. These observations are also valid when using the CT or EF settings for the 1st-stage.

7.4.6 Comparison to the State-of-the-Art

Table. 8.1 also compares the proposed two-stage OSLN detection method with 2 state-of-the-art methods, *i.e.*, the multi-task universal lesion analysis network (MULAN) [78] (achieves the best general lesion detection results in the DeepLesion dataset) and a 2.5D CNN method for classifying enlarged LNs [7] (achieves the best 2nd-stage LN classification results in the enlarged LN dataset). We retrain the MULAN using both CT and CT+PET as inputs on our radiotherapy dataset. The tagging information is

1st-Stage Setting	2nd-Stage Inputs			Evaluation Metrics	
	CT	PET	Tag	F1	mFROC
CT#				0.407	0.431
EF#		Not Applied		0.370	0.395
LF#				0.441	0.478
CT [7]	✓			0.220	0.067
CT			✓	0.380	0.408
CT	✓			0.421	0.449
CT	✓	✓		0.450	0.491
CT (GLNet)	✓	✓	✓	0.513	0.563
EF [7]	✓			0.225	0.092
EF			✓	0.397	0.444
EF	✓			0.423	0.473
EF	✓	✓		0.469	0.518
EF (GLNet)	✓	✓	✓	0.507	0.572
LF [7]	✓			0.257	0.143
LF			✓	0.471	0.531
LF	✓			0.513	0.576
LF	✓	✓		0.526	0.594
LF (GLNet)	✓	✓	✓	**0.552**	**0.645**
End-to-End Method	Inputs			Evaluation Metrics	
	CT	PET	Tag	F1	mFROC
MULAN [78]	✓	✓	✓	0.436	0.475
MULAN [78]	✓		✓	0.335	0.348

Table 7.3. Performance comparison of different methods on the testing set. The "1st-Stage Setting" column denotes which setting is used to generate OSLN candidates. "#" means we directly evaluate based on 1st-stage instance-wise segmentation scores. The "2nd-Stage Inputs" column indicates which inputs are provided to the 2nd-stage classifier. Boldface denotes our chosen 2nd-stage classifier, evaluated across different 1st-stage settings. We also compare against previous state-of-the-arts, the [7] and the end-to-end MULAN system [78].

naturally incorporated in MULAN regardless of input channels. Several conclusions can be drawn. First, MULAN's results, based on the CT+PET input (0.475 mFROC), are better than those based on the CT alone (0.348 mFROC), which again demonstrates the importance of PET imaging in the OSLN detecting task, even when using a single end-to-end trained model. Second, MULAN's best performance is just comparable with our best 1st-stage-only results, *i.e.*, (LF#). This demonstrates the effectiveness

of our 1st-stage with distance stratification and the two-stream network fusion. Third, our complete pipeline, regardless of the 1st-stage settings, significantly outperforms the best MULAN results, *e.g.*, CT (GLNet) achieves an mFROC score of 0.563 as compared to 0.475 from MULAN, whereas LF (GLNet) further boosts the mFROC to 0.645. This is a 22% improvement and highlights the advantages of our two-stage method, which is tailored to achieve maximum performance gain on the challenging and unique OSLN problem.

Similar to our 2nd-stage, the 2.5D CNN method of [7] is designed to classify LN candidates, but it was characterized only on enlarged LN candidates using contrast-enhanced CT. We trained it using our non-contrast CT local patches under different 1st-stage settings, *i.e.*, CT, EF and LF. Note that it has the worst performance among all 2nd-stage classifiers, with a best mFROC of only 0.143. This large performance degradation, particularly compared to our CT-only 2nd-stage classifier, is probably due to its 2.5D input setup and the missing of PET information. Although the 2.5D inputs and 3 orthogonal views is efficient for enlarged LN classification [7], this pseudo 3D analysis cannot fully leverage the 3D information that seems important to differentiate OSLNs from background.

7.5 Conclusion and Future Works

We proposed a new two-stage approach to automatically detect and segment oncology significant lymph nodes (OSLNs) from non-contrast CT and PET, which has not been previously studied as a computational task. In the 1st-stage, we introduce a divide-and-conquer distance stratification method by dividing OSLNs into tumor-proximal and tumor-distal categories; followed by training separate detection-by-segmentation networks to learn the category specific features aimed to decouple this challenging task into two easier ones. In the 2nd-stage, we propose the GLNet to further reduce the false positives from the 1st-stage, by combining local appearance features from

CT/PET patches and global semantic information migrated from a general lesion-characteristics-tagging model. Our method is evaluated on the largest OSLN dataset of 141 esophageal cancer patients. Our proposed framework significantly improves the recall from 45% to 67% at the 3 false-positive rates per patient as compared to previous state-of-the-art methods. Thus, our work represents an important step forward toward OSLNs detection and segmentation. In the future, we would like to consider the relationship among OSLNs and the primary tumors in the false positive reduction stage.

Chapter 8

Lymph Node Gross Tumor Volume Detection and Segmentation via Distance-based Gating using 3D CT/PET Imaging in Radiotherapy

Finding, identifying and segmenting suspicious cancer metastasized lymph nodes from 3D multi-modality imaging is a clinical task of paramount importance. In radiotherapy, they are referred to as Lymph Node Gross Tumor Volume (GTV_{LN}[1]). Determining and delineating the spread of GTV_{LN} is essential in defining the corresponding resection and irradiating regions for the downstream workflows of surgical resection and radiotherapy of various cancers. In this work, we propose an effective distance-based gating approach to simulate and simplify the high-level reasoning protocols conducted by radiation oncologists, in a divide-and-conquer manner. GTV_{LN} is divided into two subgroups of "tumor-proximal" and "tumor-distal", respectively, by means of binary or soft distance gating. This is motivated by the observation that each category can have distinct though overlapping distributions of appearance, size and other LN characteristics. A novel multi-branch detection-by-segmentation network is trained with each branch specializing on learning one GTV_{LN} category features, and outputs from multi-branch are fused in inference. The proposed method is evaluated on an in-house dataset of 141 esophageal cancer patients with both PET and CT imaging modalities. Our

results validate significant improvements on the mean recall from 72.5% to 78.2%, as compared to previous state-of-the-art work. The highest achieved GTV_{LN} recall of 82.5% at 20% precision is clinically relevant and valuable since human observers tend to have low sensitivity ($\sim$ 80% for the most experienced radiation oncologists, as reported by literature [25]).

8.1 Introduction

Assessing the lymph node (LN) status in oncology clinical workflows is an indispensable step for the precision cancer diagnosis and treatment planning, e.g., radiation therapy or surgical resection. The class of enlarged LN is defined by the revised RECIST guideline [46] if its short axial axis is more than 10-15 mm in computed tomography (CT). In radiotherapy treatment, both the primary tumor and all metastasis suspicious LNs must be sufficiently treated within the clinical target volume with the proper doses [17]. We refer to these LNs as lymph node gross tumor volume or GTV_{LN}, which includes enlarged LNs, as well as smaller ones that are associated with a high positron emission tomography (PET) signal or any metastasis signs in CT [107]. Accurately identifying and delineating GTV_{LN}, to be spatially included in the treatment area, is essential for a desirable cancer treatment outcome [54].

It is an extremely challenging and time-consuming task to identify GTV_{LN}, even for experienced radiation oncologists. High-level sophisticated clinical reasoning guidelines are needed, leading to the risk of uncertainty and subjectivity with high inter-observer variabilities [25]. It is arguably more difficult than detecting the more general enlarged LNs. (1) Finding GTV_{LN} is often performed using radiotherapy CT (RTCT) that (unlike diagnostic CT) is not contrast-enhanced. Hence the metastasis signs for identifying GTV_{LN} are subtler. (2) GTV_{LN} itself has poor contrast. Because of the

[1]Both the GTV_{LN} and the OSLNs in Chapter 7 have the same definition and refer to the same thing.

shape and appearance ambiguity, it can be easily confused with vessels or muscles. (3) The size and shape of GTV_{LN} vary considerably with large amounts of smaller ones that are harder to detect. Refer to Fig. 8.1 (top row) for an illustration of GTV_{LN}. While many previous works attempt to detect enlarged LNs using contrast-enhanced CT [2], [7], [47], [48], [51], [53], [55], no work, as of yet, has studied the GTV_{LN} detection in non-contrast RTCT scans. Given the evident differences between the enlarged LNs and GTV_{LN}, further innovations are required for the robust GTV_{LN} detection and segmentation.

Valuable insights from physicians' clinical diagnosis and analysis process can be leveraged to tackle this problem. As one of the primary cues, human observers condition the analysis of GTV_{LN} based on the LNs' distance with respect to the corresponding primary tumor location. For LNs proximal to the tumor, physicians more readily identify them as GTV_{LN} in radiotherapy treatment. However, for LNs distal to the tumor, they use more strict criteria to include if there are clear signs of metastasis, *e.g.*, enlarged size, increased PET signals, and/or other CT based evidence [107]. Hence, the distance measure relative to the primary tumor plays a key role during physician's decision making. Besides the distance, the PET modality is also of high importance. Although as a noisy imaging channel, it has shown to be helpful in increasing the GTV_{LN} detection sensitivity [25]. As demonstrated in Fig. 8.1 (bottom row), PET provides critically distinct information, yet, it also exhibits false positives (FPs) and false negatives (FNs).

In this paper, we imitate the physician's diagnosis process to tackle the problem of GTV_{LN} detection and segmentation. (1) We introduce a distance-based gating strategy in a multi-task framework to divide the underlying GTV_{LN} distributions into "tumor-proximal" and "tumor-distal" categories and solve them accordingly. Specifically, a multi-branch network is proposed to adopt a shared encoder and two separate decoders to detect and segment the "tumor-proximal" and "tumor-distal" GTV_{LN}, respectively.

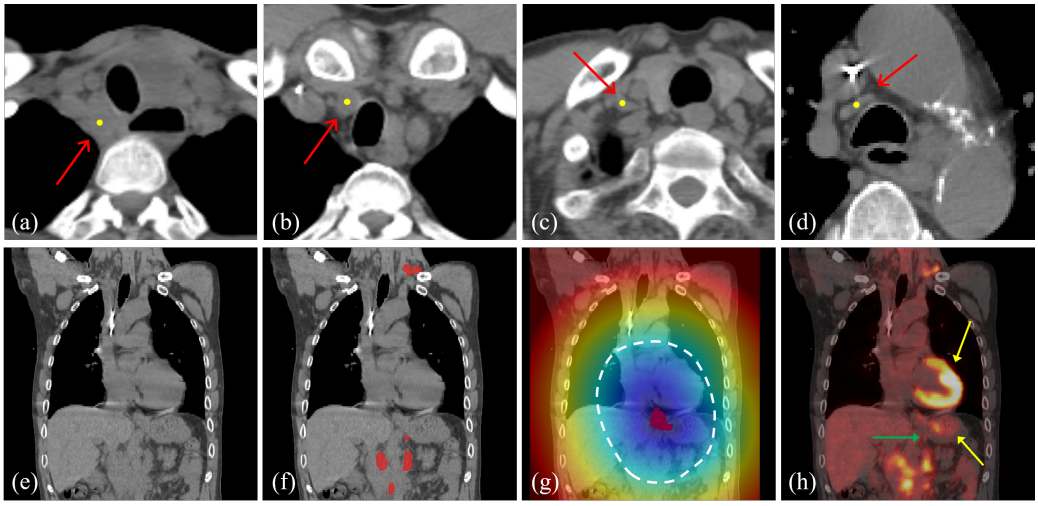

Figure 8.1. Top row (a-d): examples of the GTV_{LN} (red arrow) with varying size and appearance at scatteredly distributed locations. Bottom row (e-h): (e) A coronal view of RTCT for an esophageal cancer patient. (f) The manual annotated GTV_{LN} mask. (g) The tumor distance transformation map overlaid on RTCT, where the primary tumor is indicated by red in the center and the white dash line shows an example of the binary tumor proximal and distal region division. (h) PET imaging shows several FPs with high signals (yellow arrows). Two FN GTV_{LN} are indicated by green arrow where PET has even no signals on a GTV_{LN}.

A distance-based gating function is designed to generate the corresponding GTV_{LN} sample weights for each branch. By applying the gating function at the outputs of decoders, each branch is specialized to learn the "tumor-proximal" or "tumor-distal" GTV_{LN} features that emulates physician's diagnosis process. (2) We leverage the early fusion (EF) of three modalities as input to our model, *i.e.*, RTCT, PET and 3D tumor distance map (Fig. 8.1(bottom row)). RTCT depicts anatomical structures capturing the intensity, appearance and contextual information, while PET provides metastasis functional activities. Meanwhile, the tumor distance map further encodes the critical distance information in the network. Fusion of these three modalities together can effectively boost the GTV_{LN} identification performance. (3) We evaluate on a dataset comprising 651 voxel-wise labeled GTV_{LN} instances in 141 esophageal cancer patients, as the largest GTV_{LN} dataset to date for chest and abdominal radiotherapy. Our method significantly improves the detection mean recall from

72.5% to 78.2%, compared with the previous state-of-the-art lesion detection method [78]. The highest achieved recall of 82.5% is also clinically relevant and valuable. As reported in [25], human observers tend to have relatively low GTV_{LN} sensitivities, *e.g.*, $\sim 80\%$ by even very experienced radiation oncologists. This demonstrates our work's clinical values.

8.2 Method

Fig. 8.2 shows the framework of our proposed multi-branch GTV_{LN} detection-by-segmentation method. Similar to [20], [122] which are designed for the pancreatic tumors, we detect GTV_{LN} by segmenting them. We first compute the 3D tumor distance transformation map (Sec. 8.2.1), based on which any GTV_{LN} is divided into the tumor-proximal or tumor-adjacent subcategory. Next, a multi-branch detection-by-segmentation network is designed where each branch focuses on one subgroup of GTV_{LN} segmentation (Sec. 8.2.2). This is achieved by applying a binary or soft distance-gating function imposed on the penalty function at the output of the two branches (Sec. 8.2.3). Hence, each branch can learn specific parameters to specialize on segmenting and detecting the tumor-proximal and tumor-adjacent GTV_{LN}, respectively.

8.2.1 3D Tumor Distance Transformation

To stratify GTV_{LN} into tumor-proximal and tumor-distal subgroups, we first compute the 3D tumor distance transformation map, denoted as $\mathbf{X}^{\text{D}}$, from the primary tumor $\mathcal{O}$. The value at each voxel x_i represents the shortest distance between this voxel and the mask of the primary tumor. Let $B(\mathcal{O})$ be a set that includes the boundary voxels of the tumor. The distance transformation value at a voxel x_i is computed as

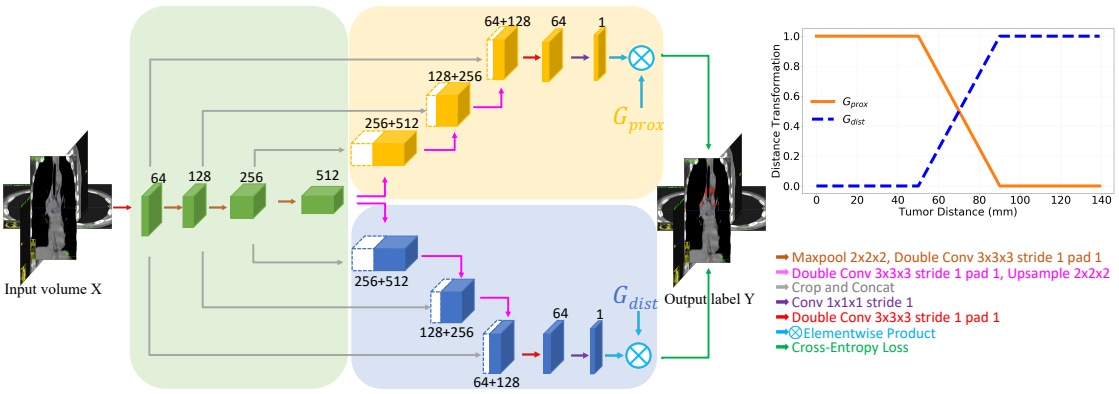

Figure 8.2. The overall framework of our proposed multi-branch GTV_{LN} detection and segmentation method. The light green part shows the encoder path, while the light yellow and light blue parts show the two decoders, respectively. The number of channels is denoted either on the top or the bottom of the box.

$$\mathbf{X}^{\text{D}}(x_i) = \begin{cases} \min\limits_{q \in B(\mathcal{O})} d(x_i, q) & \text{if} \quad x_i \notin \mathcal{O} \\ 0 & \text{if} \quad x_i \in \mathcal{O} \end{cases}, \tag{8.1}$$

where $d(x_i, q)$ is the Euclidean distance from x_i to q. $\mathbf{X}^{\text{D}}$ can be efficiently computed using algorithms such as the one proposed in [116]. Based on $\mathbf{X}^{\text{D}}$, GTV_{LN} can be divided into tumor-proximal and tumor-distal subgroups using either binary or soft distance-gating function as explained in detail in Sec. 8.2.3.

8.2.2 Multi-branch Detection-by-Segmentation via Distance Gating

GTV_{LN} identification is implicitly associated with their distance distributions to the primary tumor in the diagnosis process of physicians. Hence, we divide GTV_{LN} into tumor-proximal and tumor-distal subgroups and conduct detection accordingly. To do this, we design a multi-branch detection-by-segmentation network with each branch focusing on segmenting one GTV_{LN} subgroup. Each branch is implemented by an independent decoder to learn and extract the subgroup specific information, while they share a single encoder to extract the common GTV_{LN} image features. Assuming

there are N data samples, we denote a dataset as $\mathbf{S} = \left\{\left(\mathbf{X}_n^{\mathrm{CT}}, \mathbf{X}_n^{\mathrm{PET}}, \mathbf{X}_n^{\mathrm{D}}, \mathbf{Y}_n\right)\right\}_{n=1}^{N}$, where $\mathbf{X}_n^{\mathrm{CT}}$, $\mathbf{X}_n^{\mathrm{PET}}$, $\mathbf{X}_n^{\mathrm{D}}$ and $\mathbf{Y}_n$ represent the non-contrast RTCT, registered PET, tumor distance transformation map, and ground truth GTV_{LN} segmentation mask, respectively. Without the loss of generality, we drop n for conciseness in the rest of this paper. The total number of branches is denoted as M, where $M = 2$ in our case. A CNN segmentation model is denoted as a mapping function $\mathbb{E} : \mathbf{P} = \mathbf{f}(\mathcal{X}; \mathbf{\Theta})$, where $\mathcal{X}$ is a set of inputs, which consists of a single modality or a concatenation of multiple modalities. $\mathbf{\Theta}$ indicates model parameters, and $\mathbf{P}$ means the predicted probability volume. Given that $p(y_i|x_i; \mathbf{\Theta}_m)$ represents the predicted probability of a voxel $x_i \in \mathcal{X}$ being the labeled class from the mth branch, the overall negative log-likelihood loss aggregated across M branches can be formulated as:

$$\mathcal{L} = \sum_m \mathcal{L}_m(\mathcal{X}; \mathbf{\Theta}_m, \boldsymbol{G}_m) = -\sum_i \sum_m g_{m,i} \log(p(y_i|x_i; \mathbf{\Theta}_m)), \tag{8.2}$$

where $\boldsymbol{G} = \{\boldsymbol{G}_m\}_{m=1}^{M}$ is introduced as a set of volumes containing the transformed gating weights at each voxel based on its distance to the primary tumor. At every voxel $x_i \in \boldsymbol{G}$, the gating weights satisfies $\sum_m g_{m,i} = 1$.

8.2.3 Distance-based Gating Module

Based on the tumor distance map $\mathbf{X}^{\mathrm{D}}$, our gating functions can be designed to generate appropriate GTV_{LN} sample weights for different branches so that each branch specializes on learning the subgroup specific features. In our case, we explore two options: (1) binary distance gating and (2) soft distance gating.

Binary Distance Gating (BG). Based on the tumor distance map $\mathbf{X}^{\mathrm{D}}$, we divide image voxels into two groups, x_{prox} and x_{dis}, to be tumor-proximal and tumor-distal, respectively, where $\mathrm{prox} = \{i|x_i^{\mathrm{D}} \leq d_0, x_i^{\mathrm{D}} \in \mathbf{X}^{\mathrm{D}}\}$ and $\mathrm{dis} = \{i|x_i^{\mathrm{D}} > d_0, x_i^{\mathrm{D}} \in \mathbf{X}^{\mathrm{D}}\}$. Therefore the gating transformations for two decoders are defined as $\boldsymbol{G}_{\mathbf{prox}} = \mathbf{1}[x_i^{\mathrm{D}} \leq d_0]$ and $\boldsymbol{G}_{\mathbf{dist}} = 1 - \boldsymbol{G}_{\mathbf{prox}}$, where $\mathbf{1}[\cdot]$ is an indicator function which equals one if its

argument is true and zero otherwise. In this way, we divide the GTV_{LN} strictly into two disjoint categories, and each branch focuses on decoding and learning from one category.

Soft Distance Gating (SG). We further explore a soft gating method that linearly changes the penalty weights of GTV_{LN} samples as they are closer or further to the tumor. This can avoid a sudden change of weight values when samples are near the proximal and distal category boundaries. Recommended by our physician, we formulate a soft gating module based on $\mathbf{X}^{\text{D}}$ as following:

$$\mathbf{G_{prox}}(x_i) = \begin{cases} 1 - \frac{x_i^{\text{D}} - d_{prox}}{d_{dist} - d_{prox}} & \text{if} \quad d_{prox} < x_i^{\text{D}} \leq d_{dist} \\ 1 & \text{if} \quad x_i^{\text{D}} \leq d_{prox} \\ 0 & \text{if} \quad x_i^{\text{D}} > d_{dist} \end{cases}, \tag{8.3}$$

and $\mathbf{G_{dist}}(x_i) = 1 - \mathbf{G_{prox}}(x_i)$ accordingly.

8.3 Experiments

8.3.1 Dataset and Preprocessing

Dataset. We collected 141 non-contrast RTCTs of esophageal cancer patients, with all undergoing radiotherapy treatments. Radiation oncologists labeled 3D segmentation masks of the primary tumor and all GTV_{LN}. For each patient, we have a non-contrast RTCT and a pair of PET/CT scans. There is a total of 651 GTV_{LN} with voxel-wise annotations in the mediastinum or upper abdomen regions, as the largest annotated GTV_{LN} dataset to-date. We randomly split patients into 60%, 10%, 30% for training, validation and testing, respectively.

Implementation Details. In our experiments, PET scan is registered to RTCT using the similar method described in [113]. Then all coupling pairs of RTCT and registered PET images are resampled to have a consistent spatial resolution of $1 \times 1 \times 2.5$ mm. To generate the 3D training samples, we crop sub-volumes of $96 \times 96 \times 64$ from the RTCT, registered PET and the tumor distance map around each GTV_{LN} as well

as randomly from the background. For the distance-gating related parameters, we set $d_0 = 7$ cm as the binary gating threshold, and $d_{prox} = 5$ cm and $d_{dist} = 9$ cm as the soft gating thresholds, respectively, as suggested by our clinical collaborator. We further apply random rotations in the x-y plane within 10 degrees to augment the training data.

Detection-by-segmentation models are trained on two NVIDIA Quadra RTX 6000 GPUs with a batch size of 8 for 50 epochs. The RAdam [117] optimizer with a learning rate of 0.0001 is used with a momentum of 0.9 and a weight decay of 0.0005. For inference, 3D sliding windows with a sub-volume of $96 \times 96 \times 64$ and a stride of $64 \times 64 \times 32$ voxels are processed. For each sub-volume, predictions from two decoders are weighted and aggregated according to the gating transformation $\mathbf{G_m}$ to obtain the final GTV_{LN} segmentation results.

Evaluation Metrics. We first describe the hit criteria, *i.e.*, the correct detection, for our detection-by-segmentation method. For an GTV_{LN} prediction, if it overlaps with any ground-truth GTV_{LN}, we treat it as a hit provided that its estimated radius is similar to the radius of the ground-truth GTV_{LN} within the range of $[0.5, 1.5]$. The performance is assessed using the mean and max recall (mRecall and Recall_{max}) at a precision range of $[0.10, 0.50]$ with 0.05 interval, and the mean free response operating characteristic (FROC) at $3, 4, 6, 8$ FPs per patient. These operating points were chosen after confirming with our physician.

Comparison Setups. Using the binary and soft distance-based gating function, our multi-branch GTV_{LN} detection-by-segmentation method is denoted as **multi-branch BG** and **multi-branch SG**, respectively. We compare against the following setups: (1) a single 3D UNet [10] trained using RTCT alone or the early fusion (EF) of multi-modalities (denoted as **single-net** method); (2) Two separate UNets trained with the corresponding tumor-proximal and tumor-distal GTV_{LN} samples and results spatially fused together (our preliminary work [123] denoted as **multi-net**

Methods:	CT	EF	mRecall	Recall_{max}	mFROC	FROC@4	FROC@6
single-net	✓		0.664	0.762	0.604	0.552	0.675
single-net		✓	0.731	0.820	0.676	0.667	0.713
multi-net BG [123]		✓	0.747	0.825	0.695	0.668	**0.739**
multi-branch BG (Ours)		✓	0.761	**0.845**	0.679	0.667	0.716
multi-branch SG (Ours)		✓	**0.782**	0.843	**0.724**	**0.729**	0.738
MULAN [78]	✓		0.711	0.758	0.632	0.632	0.642
MULAN [78]		✓	0.725	0.781	0.708	0.718	0.720

Table 8.1. Quantitative results of our proposed methods with the comparison to other setups and the previous state-of-the-art.

BG); and (3) MULAN [78], a state-of-the-art (SOTA) general lesion detection method on DeepLesion [2] that contains more than 10,000 enlarged LNs.

8.3.2 Quantitative Results & Discussion

Our quantitative results and comparisons are given in Table. 8.1. Several observations can be drawn on addressing the effectiveness of our proposed methods. **(1)** The multi-modality input, *i.e.*, early fusion (EF) of RTCT, PET and tumor distance map, are of great benefits for detecting the GTV_{LN}. There are drastic performance improvements of absolute 6.7% and 7.2% in mRecall and mFROC when EF is adopted as compared to using RTCT alone. These results validate that input channels of PET functional imaging and 3D tumor distance transform map are valuable for identifying GTV_{LN}. **(2)** The distance-based gating strategies are evidently effective as the options of **multi-net BG**, **multi-branch BG** and **multi-branch SG** consistently increase the performance. For example, the multi-net BG model achieves 74.7% mRecall and 69.5% mFROC, which is a 1.6% and 1.9% improvement against the best single-net model (where no distance-based stratification is used). The performance further boosts with the network models of multi-branch BG and multi-branch SG, to the highest scores of 78.2% mRecall and 72.4% mFROC achieved by the multi-branch SG.

Multi-branch versus Multi-net. Using the distance-based gating strategy, our proposed multi-branch methods perform considerably better than the **multi-net BG**

model. Even our second best model **multi-branch BG**, the mean and maximal recalls have been improved by 1.4% (from 74.7% to 76.1%) and 2.0% (from 82.5% to 84.5%) against the **multi-net BG** model. When the multi-branch framework is equipped with the **soft-gating**, marked improvements of absolute 3.5% and 2.9% in both mRecall and mFROC are observed as compared against to the **multi-net BG** model. This validates the effectiveness of our jointly trained multi-branch framework design, and our intuition that gradually changing GTV_{LN} weights for the proximal and distal branches are more natural and effective. As we recall, the multi-net baseline directly trains two separate 3D UNets [10] targeted to segment each GTV_{LN} subgroup. Considering the limited GTV_{LN} training data (a few hundreds of patients), it can be overfitting prone from the split to even smaller patient subgroups.

Table. 8.1 also compares with the SOTA universal lesion detection method, i.e., MULAN [78] on DeepLesion [2], [110]. We have retrained the MULAN models using both CT and EF inputs, but even the best results, *i.e.*, using EF, have a large gap (72.5% vs. 78.2% mRecall) with our distance-gating networks, which further proves that the tumor distance transformation cue plays a key role in GTV_{LN} identification.

Fig. 8.3 illustrates the visualization results of our method compared to other baselines. For the enlarged GTV_{LN} (top row), most methods can detect it correctly. However, as the size of GTV_{LN} becomes smaller and the contrast is poorer, our method can still successfully detect them while others struggled.

8.4 Conclusion and Future Works

In this work, we propose an effective distance-based gating approach in a multi-task deep learning framework to segment GTV_{LN}, emulating the oncologists' high-level diagnosis protocols. GTV_{LN} is divided into two subgroups of "tumor-proximal" and "tumor-distal", by means of binary or soft distance gating. A novel multi-branch

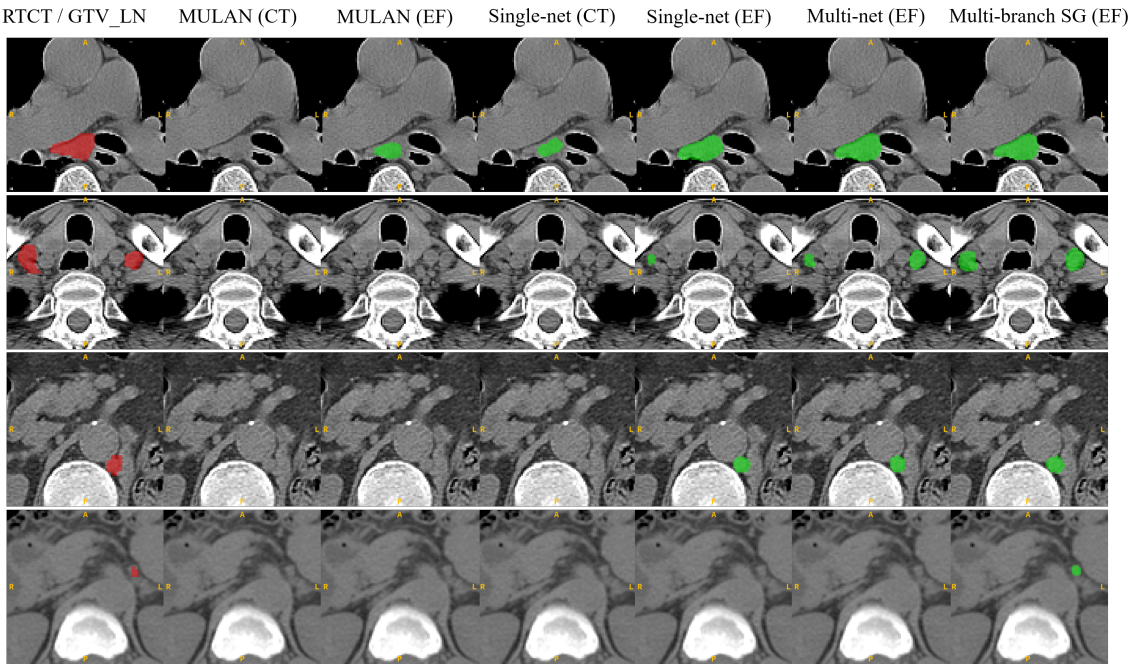

Figure 8.3. Four qualitative examples of the detection results using different methods. Red color represents the ground-truth GTV_{LN} overlaid on the RTCT images; Green color indicates the predicted segmentation masks. As shown, for the enlarged GTV_{LN} (top row), most methods can detect it correctly. However, as GTV_{LN} size becomes smaller and contrast is poor, our method can successfully detect them while others struggled.

detection-by-segmentation network is trained with each branch specializing on learning one subgroup features. We evaluate our method on a dataset of 141 esophageal cancer patients. Our results demonstrate significant performance improvements on the mean recall from 72.5% to 78.2%, as compared to previous state-of-the-art work. The highest achieved GTV_{LN} recall of 82.5% at the 20% precision level is clinically relevant and valuable. Future works can include finding lymph node station firsts and then segment GTV_{LN} afterwards. Another direction worth considering is to replace the Euclidean distance map with the geodesic anatomic distance inside the lymphatic system, which is more realistic.

Chapter 9

Conclusion and Future Work

In this dissertation, we focus on the topic of improving the state diagnostic medicine by advancing the volumetric medical image segmentation with state-of-the-art deep learning techniques. Chapters from Chapter 3 to Chapter 6 are in the scope of segmenting organs/tumors more accurate while Chapter 7 and Chapter 8 fall into addressing the detection by segmentation of metastasis-suspicious lymph nodes. In Chapter 3, we propose the ResDSN network backbone in a general 3D coarse-to-fine framework to tackle the challenges of limited amount of annotated data and limited computational resources in volumetric medical image segmentation. The quantitative results show that the proposed methods can improve the overall segmentation accuracy as well as the segmentation in the worst case. Chapter 4 and Chapter 5 extend Chapter 3 to segment lethal pancreatic tumors, *i.e.*, PDAC and PNETs, respectively, which achieve clinically promising sensitivity and specificity. Moving forward, Chapter 6 explores the novel idea of AutoML in the medical imaging field to automatically search the network architectures tailing for the volumetric medical image segmentation whereas almost all prior works adopt human-designed network backbones. Moving forward beyond normal organs and tumors segmentation, Chapter 7 is the first computationally realization of detecting, identifying and characterizing suspicious cancer metastasized lymph nodes by proposing a 3D distance stratification strategy to simulate and simplify the high-level reasoning protocols conducted by radiation

oncologists in a divide-and-conquer manner. Our experiments indicate that the local textures and global semantically meaningful tagging features can largely improve the lymph node detection accuracy. Chapter 8 upgrades the distance-based stratification by a multi-branch detection-by-segmentation network, which further advances the finding, identifying and segmenting of metastasis-suspicious lymph nodes.

We demonstrate that designing/searching advanced network backbone/architectures and/or migrating rich knowledge from routine works of medical experts into automatic computer-aided diagnosis solutions can improve both medical diagnosis. Further works could lie in many directions. First, in a sense that medical image annotation would always be limited in the scope of deep learning, segmentation with scarce training data (few shot learning), with image-level annotation (weakly supervised learning), and with a mixture of labelled and unlabelled data (semi-supervised learning) are important directions. Second, in real applications, computer-aided diagnosis (CAD) solutions need to be applied to multiple sites, with each site have different protocols, scanners, population distribution and *etc*. So domain adaptation [124] would be the direction to mitigate the data discrepancy across different sites. Third, when CAD solutions are deployed locally, how to have medical experts in the loop would be an open question. For example, when medical experts are using the CAD system, how the feedback of human assessment improves the CAD and how the CAD improves the human routine works remain challenging. Ideally, the machine intelligence and human beings iteratively evolve into a win-win situation.

References

[1] L. Lu, A. Barbu, M. Wolf, J. Liang, M. Salganicoff, and D. Comaniciu, "Accurate polyp segmentation for 3d ct colongraphy using multi-staged probabilistic binary learning and compositional model," in *CVPR*, 2008.

[2] K. Yan, X. Wang, L. Lu, and R. M. Summers, "Deeplesion: Automated mining of large-scale lesion annotations and universal lesion detection with deep learning," *Journal of Medical Imaging*, vol. 5, no. 3, p. 036 501, 2018.

[3] L. Lu and A. P. Harrison, "Deep medical image computing in preventive and precision medicine," *IEEE MultiMedia*, vol. 25, no. 3, pp. 109–113, 2018.

[4] P. Kickingereder, F. Isensee, I. Tursunova, J. Petersen, U. Neuberger, D. Bonekamp, G. Brugnara, M. Schell, T. Kessler, M. Foltyn, *et al.*, "Automated quantitative tumour response assessment of mri in neuro-oncology with artificial neural networks: A multicentre, retrospective study," *The Lancet Oncology*, vol. 20, no. 5, pp. 728–740, 2019.

[5] S. Miao, Z. J. Wang, and R. Liao, "A cnn regression approach for real-time 2d/3d registration," *TMI*, vol. 35, no. 5, pp. 1352–1363, 2016.

[6] X. Yang, R. Kwitt, and M. Niethammer, "Fast predictive image registration," in *Deep Learning and Data Labeling for Medical Applications*, Springer, 2016, pp. 48–57.

[7] H. R. Roth, L. Lu, J. Liu, J. Yao, A. Seff, K. Cherry, L. Kim, and R. M. Summers, "Improving computer-aided detection using convolutional neural networks and random view aggregation," *TMI*, vol. 35, no. 5, pp. 1170–1181, 2016.

[8] W. Shen, M. Zhou, F. Yang, C. Yang, and J. Tian, "Multi-scale convolutional neural networks for lung nodule classification," in *IPMI*, 2015.

[9] Q. Dou, H. Chen, L. Yu, L. Zhao, J. Qin, D. Wang, V. C. Mok, L. Shi, and P.-A. Heng, "Automatic detection of cerebral microbleeds from mr images via 3d convolutional neural networks," *TMI*, vol. 35, no. 5, pp. 1182–1195, 2016.

[10] Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger, "3D u-net: Learning dense volumetric segmentation from sparse annotation," in *MICCAI*, 2016.

[11] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-net: Fully convolutional neural networks for volumetric medical image segmentation," in *3DV*, 2016.

[12] X. Liu, H. R. Tizhoosh, and J. Kofman, "Generating binary tags for fast medical image retrieval based on convolutional nets and radon transform," in *IJCNN*, 2016.

[13] D. Yang, D. Xu, S. K. Zhou, B. Georgescu, M. Chen, S. Grbic, D. Metaxas, and D. Comaniciu, "Automatic liver segmentation using an adversarial image-to-image network," in *MICCAI*, 2017.

[14] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. Van Der Laak, B. Van Ginneken, and C. I. Sánchez, "A survey on deep learning in medical image analysis," *MIA*, vol. 42, pp. 60–88, 2017.

[15] M. Goyal, M. H. Yap, N. D. Reeves, S. Rajbhandari, and J. Spragg, "Fully convolutional networks for diabetic foot ulcer segmentation," in *2017 IEEE international conference on systems, man, and cybernetics (SMC)*, IEEE, 2017, pp. 618–623.

[16] K.-L. Liu, T. Wu, P.-T. Chen, Y. M. Tsai, H. Roth, M.-S. Wu, W.-C. Liao, and W. Wang, "Deep learning to distinguish pancreatic cancer tissue from non-cancerous pancreatic tissue: A retrospective study with cross-racial external validation," *The Lancet Digital Health*, vol. 2, no. 6, e303–e313, 2020.

[17] D. Jin, D. Guo, T.-Y. Ho, A. P. Harrison, J. Xiao, C.-K. Tseng, and L. Lu, "Deep esophageal clinical target volume delineation using encoded 3d spatial context of tumors, lymph nodes, and organs at risk," in *MICCAI*, 2019.

[18] H. R. Roth, L. Lu, A. Farag, H.-C. Shin, J. Liu, E. B. Turkbey, and R. M. Summers, "Deeporgan: Multi-level deep convolutional networks for automated pancreas segmentation," in *MICCAI*, 2015.

[19] A. L. Simpson, M. Antonelli, S. Bakas, M. Bilello, K. Farahani, B. van Ginneken, A. Kopp-Schneider, B. A. Landman, G. Litjens, B. Menze, *et al.*, "A large annotated medical image dataset for the development and evaluation of segmentation algorithms," *arXiv:1902.09063*, 2019.

[20] Z. Zhu, Y. Xia, L. Xie, E. K. Fishman, and A. L. Yuille, "Multi-scale coarse-to-fine segmentation for screening pancreatic ductal adenocarcinoma," in *MICCAI*, 2019.

[21] P. C. UK, *Pancreatic cancer uk*, 2019.

[22] B. W. K. P. Stewart, C. P. Wild, *et al.*, "World cancer report 2014," *Health*, 2017.

[23] PDQ Adult Treatment Editorial Board, "Pancreatic cancer treatment (PDQ®),"

[24] Z. Zhu, D. Jin, K. Yan, T.-Y. Ho, X. Ye, D. Guo, C.-H. Chao, J. Xiao, A. Yuille, and L. Lu, "Lymph node gross tumor volume detection and segmentation via distance-based gating using 3d ct/pet imaging in radiotherapy," in *MICCAI*, 2020.

[25] R. Goel, W. Moore, B. Sumer, S. Khan, D. Sher, and R. M. Subramaniam, "Clinical practice in pet/ct for the management of head and neck squamous cell cancer," *American Journal of Roentgenology*, vol. 209, no. 2, pp. 289–303, 2017.

[26] L. C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," in *ICLR*, 2016.

[27] S. Xie and Z. Tu, "Holistically-nested edge detection," in *ICCV*, 2015.

[28] Z. Zhu, X. Wang, S. Bai, C. Yao, and X. Bai, "Deep learning representation using autoencoder for 3D shape retrieval," *Neurocomputing*, vol. 204, pp. 41–50, 2016.

[29] H. R. Roth, L. Lu, A. Farag, A. Sohn, and R. M. Summers, "Spatial aggregation of holistically-nested networks for automated pancreas segmentation," in *MICCAI*, 2016.

[30] Y. Zhou, L. Xie, W. Shen, Y. Wang, E. K. Fishman, and A. L. Yuille, "A fixed-point model for pancreas segmentation in abdominal CT scans," in *MICCAI*, 2017.

[31] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *CVPR*, 2015.

[32] T. D. Bui, J. Shin, and T. Moon, "3D densely convolution networks for volumetric segmentation," *arXiv:1709.03199*, 2017.

[33] H. Chen, Q. Dou, L. Yu, J. Qin, and P.-A. Heng, "Voxresnet: Deep voxelwise residual networks for brain segmentation from 3d mr images," *NeuroImage*, vol. 170, pp. 446–455, 2018.

[34] L. Yu, J.-Z. Cheng, Q. Dou, X. Yang, H. Chen, J. Qin, and P.-A. Heng, "Automatic 3D cardiovascular MR segmentation with densely-connected volumetric convnets," in *MICCAI*, 2017.

[35] J. Cai, L. Lu, Y. Xie, F. Xing, and L. Yang, "Improving deep pancreas segmentation in CT and MRI images via recurrent neural contextual learning and direct loss function," in *MICCAI*, 2017.

[36] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *MICCAI*, 2015.

[37] Q. Dou, L. Yu, H. Chen, Y. Jin, X. Yang, J. Qin, and P.-A. Heng, "3d deeply supervised network for automated segmentation of volumetric medical images," *MIA*, vol. 41, pp. 40–54, 2017.

[38] Z. Zhu, Y. Xia, W. Shen, E. K. Fishman, and A. L. Yuille, "A 3d coarse-to-fine framework for volumetric medical image segmentation," in *3DV*, 2018.

[39] Z. Qiu, T. Yao, and T. Mei, "Learning spatio-temporal representation with pseudo-3d residual networks," in *ICCV*, 2017.

[40] S. Liu, D. Xu, S. K. Zhou, O. Pauly, S. Grbic, T. Mertelmeier, J. Wicklein, A. Jerebko, W. Cai, and D. Comaniciu, "3d anisotropic hybrid network: Transferring convolutional features from 2d images to 3d anisotropic volumes," in *MICCAI*, 2018.

[41] G. Wang, W. Li, S. Ourselin, and T. Vercauteren, "Automatic brain tumor segmentation using cascaded anisotropic convolutional neural networks," in *MICCAI Brainlesion Workshop*, 2017.

[42] C. Liu, L.-C. Chen, F. Schroff, H. Adam, W. Hua, A. Yuille, and L. Fei-Fei, "Auto-deeplab: Hierarchical neural architecture search for semantic image segmentation," in *CVPR*, 2019.

[43] H. Liu, K. Simonyan, and Y. Yang, "Darts: Differentiable architecture search," in *ICLR*, 2019.

[44] A. Mortazi and U. Bagci, "Automatically designing cnn architectures for medical image segmentation," in *International Workshop on MLMI*, 2018.

[45] E. A. Eisenhauer, P. Therasse, J. Bogaerts, L. H. Schwartz, D. Sargent, R. Ford, J. Dancey, S. Arbuck, S. Gwyther, M. Mooney, *et al.*, "New response evaluation criteria in solid tumours: Revised recist guideline (version 1.1)," *European journal of cancer*, vol. 45, no. 2, pp. 228–247, 2009.

[46] L. Schwartz, J. Bogaerts, R. Ford, L. Shankar, P. Therasse, S. Gwyther, and E. Eisenhauer, "Evaluation of lymph nodes with recist 1.1," *European journal of cancer*, vol. 45, no. 2, pp. 261–267, 2009.

[47] D. Bouget, A. Jørgensen, G. Kiss, H. O. Leira, and T. Langø, "Semantic segmentation and detection of mediastinal lymph nodes and anatomical structures in ct data for lung cancer staging," *IJCARS*, pp. 1–10, 2019.

[48] J. Feulner, S. K. Zhou, M. Hammon, J. Hornegger, and D. Comaniciu, "Lymph node detection and segmentation in chest ct data using discriminative learning and a spatial prior," *MIA*, vol. 17, no. 2, pp. 254–270, 2013.

[49] T. Kitasaka, Y. Tsujimura, Y. Nakamura, K. Mori, Y. Suenaga, M. Ito, and S. Nawano, "Automated extraction of lymph nodes from 3-d abdominal ct images using 3-d minimum directional difference filter," in *MICCAI*, 2007.

[50] J. Liu, J. Hoffman, J. Zhao, J. Yao, L. Lu, L. Kim, E. B. Turkbey, and R. M. Summers, "Mediastinal lymph node detection and station mapping on chest ct using spatial priors and random forest," *Medical physics*, vol. 43, no. 7, pp. 4362–4374, 2016.

[51] I. Nogues, L. Lu, X. Wang, H. Roth, G. Bertasius, N. Lay, J. Shi, Y. Tsehay, and R. M. Summers, "Automatic lymph node cluster segmentation using holistically-nested neural networks and structured optimization in ct images," in *MICCAI*, 2016.

[52] H. Oda, H. R. Roth, K. K. Bhatia, M. Oda, T. Kitasaka, S. Iwano, H. Homma, H. Takabatake, M. Mori, H. Natori, *et al.*, "Dense volumetric detection and segmentation of mediastinal lymph nodes in chest ct images," in *SPIE*, 2018.

[53] H. R. Roth, L. Lu, A. Seff, K. M. Cherry, J. Hoffman, S. Wang, J. Liu, E. Turkbey, and R. M. Summers, "A new 2.5 d representation for lymph node detection using random sets of deep convolutional neural network observations," in *MICCAI*, 2014.

[54] N. C. C. Network, "Nccn clinical practice guidelines:head and neck cancers," *American Journal of Roentgenology*, vol. version 2, 2020.

[55] A. Barbu, M. Suehling, X. Xu, D. Liu, S. K. Zhou, and D. Comaniciu, "Automatic detection and segmentation of lymph nodes from ct data," *TMI*, vol. 31, no. 2, pp. 240–250, 2011.

[56] R. Pohle and K. D. Toennies, "Segmentation of medical images using adaptive region growing," in *Medical Imaging: Image Processing*, 2001.

[57] M. U. Akram and S. A. Khan, "Multilayered thresholding-based blood vessel segmentation for screening of diabetic retinopathy," *Engineering with computers*,

[58] S. S. Chandra, Y. Xia, C. Engstrom, S. Crozier, R. Schwarz, and J. Fripp, "Focused shape models for hip joint segmentation in 3d magnetic resonance images," *MIA*, vol. 18, no. 3, pp. 567–578, 2014.

[59] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *NIPS*, 2012.

[60] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv:1409.1556*, 2014.

[61] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016.

[62] J. Merkow, A. Marsden, D. Kriegman, and Z. Tu, "Dense volume-to-volume vascular boundary detection," in *MICCAI*, 2016.

[63] E. Gibson, F. Giganti, Y. Hu, E. Bonmati, S. Bandula, K. Gurusamy, B. Davidson, S. P. Pereira, M. J. Clarkson, and D. C. Barratt, "Automatic multi-organ segmentation on abdominal ct with dense v-networks," *TMI*, vol. 37, no. 8, pp. 1822–1834, 2018.

[64] Y. Xia, L. Xie, F. Liu, Z. Zhu, E. K. Fishman, and A. L. Yuille, "Bridging the gap between 2d and 3d organ segmentation," in *MICCAI*, 2018.

[65] X. Li, H. Chen, X. Qi, Q. Dou, C.-W. Fu, and P.-A. Heng, "H-denseunet: Hybrid densely connected unet for liver and tumor segmentation from ct volumes," *TMI*, vol. 37, no. 12, pp. 2663–2674, 2018.

[66] F. Isensee, P. F. Jäger, S. A. Kohl, J. Petersen, and K. H. Maier-Hein, "Automated design of deep learning methods for biomedical image segmentation," *arXiv:1904.08128*, 2019.

[67] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le, "Learning transferable architectures for scalable image recognition," in *CVPR*, 2018.

[68] E. Real, A. Aggarwal, Y. Huang, and Q. V. Le, "Regularized evolution for image classifier architecture search," in *AAAI*, 2019.

[69] C. Liu, B. Zoph, M. Neumann, J. Shlens, W. Hua, L.-J. Li, L. Fei-Fei, A. L. Yuille, J. Huang, and K. Murphy, "Progressive neural architecture search," in *ECCV*, 2018.

[70] R. Shin, C. Packer, and D. Song, "Differentiable neural network architecture search," in *ICLR (Workshop)*, 2018.

[71] E. Real, S. Moore, A. Selle, S. Saxena, Y. L. Suematsu, J. Tan, Q. V. Le, and A. Kurakin, "Large-scale evolution of image classifiers," in *ICML*, 2017.

[72] L. Xie and A. L. Yuille, "Genetic CNN," in *ICCV*, 2017.

[73] B. Zoph and Q. V. Le, "Neural architecture search with reinforcement learning," in *ICLR*, 2017.

[74] H. Cai, L. Zhu, and S. Han, "Proxylessnas: Direct neural architecture search on target task and hardware," *arXiv:1812.00332*, 2018.

[75] X. Zhang, Z. Huang, and N. Wang, "You only search once: Single shot neural architecture search via direct sparse optimization," *CoRR*, vol. abs/1811.01567, 2018.

[76] Z. Li, S. Zhang, J. Zhang, K. Huang, Y. Wang, and Y. Yu, "Mvp-net: Multi-view fpn with position-aware attention for deep universal lesion detection," in *MICCAI*, 2019.

[77] K. Yan, M. Bagheri, and R. M. Summers, "3d context enhanced region-based convolutional neural network for end-to-end lesion detection," in *MICCAI*, 2018.

[78] K. Yan, Y. Tang, Y. Peng, V. Sandfort, M. Bagheri, Z. Lu, and R. M. Summers, "Mulan: Multitask universal lesion analysis network for joint lesion detection, tagging, and segmentation," in *MICCAI*, 2019.

[79] M. Zlocha, Q. Dou, and B. Glocker, "Improving retinanet for ct lesion detection with dense masks from weak recist labels," *arXiv:1906.02283*, 2019.

[80] J. Ding, A. Li, Z. Hu, and L. Wang, "Accurate pulmonary nodule detection in computed tomography images using deep convolutional neural networks," in *MICCAI*, 2017.

[81] M. Ghafoorian, N. Karssemeijer, T. Heskes, M. Bergkamp, J. Wissink, J. Obels, K. Keizer, F.-E. de Leeuw, B. van Ginneken, E. Marchiori, *et al.*, "Deep multi-scale location-aware 3d convolutional neural networks for automated detection of lacunes of presumed vascular origin," *NeuroImage: Clinical*, vol. 14, pp. 391–399, 2017.

[82] A. A. A. Setio, F. Ciompi, G. Litjens, P. Gerke, C. Jacobs, S. J. Van Riel, M. M. W. Wille, M. Naqibullah, C. I. Sánchez, and B. van Ginneken, "Pulmonary nodule detection in ct images: False positive reduction using multi-view convolutional networks," *TMI*, vol. 35, no. 5, pp. 1160–1169, 2016.

[83] A. Teramoto, H. Fujita, O. Yamamuro, and T. Tamaki, "Automated detection of pulmonary nodules in pet/ct images: Ensemble false-positive reduction using a convolutional neural network technique," *Medical physics*, vol. 43, no. 6Part1, pp. 2821–2827, 2016.

[84] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *ICCV*, 2017.

[85] W. Shen, X. Wang, Y. Wang, X. Bai, and Z. Zhang, "Deepcontour: A deep convolutional feature learned by positive-sharing loss for contour detection," in *CVPR*, 2015.

[86] W. Shen, B. Wang, Y. Jiang, Y. Wang, and A. L. Yuille, "Multi-stage multi-recursive-input fully convolutional networks for neuronal boundary detection," in *ICCV*, 2017, pp. 2410–2419.

[87] Y. Fang, J. Xie, G. Dai, M. Wang, F. Zhu, T. Xu, and E. Wong, "3D deep shape descriptor," in *CVPR*, 2015.

[88] Y. Zhou, L. Xie, E. K. Fishman, and A. L. Yuille, "Deep supervision for pancreatic cyst segmentation in abdominal ct scans," in *MICCAI*, 2017.

[89] M. Havaei, A. Davy, D. Warde-Farley, A. Biard, A. C. Courville, Y. Bengio, C. Pal, P.-M. Jodoin, and H. Larochelle, "Brain tumor segmentation with deep neural networks," *MIA*, vol. 35, pp. 18–31, 2017.

[90] P. Moeskops, J. M. Wolterink, B. H. van der Velden, K. G. Gilhuijs, T. Leiner, M. A. Viergever, and I. Išgum, "Deep learning for multi-task medical image segmentation in multiple modalities," in *MICCAI*, 2016.

[91] H. Roth, M. Oda, N. Shimizu, H. Oda, Y. Hayashi, T. Kitasaka, M. Fujiwara, K. Misawa, and K. Mori, "Towards dense volumetric pancreas segmentation in CT using 3D fully convolutional networks," in *SPIE*, 2018.

[92] C.-Y. Lee, S. Xie, P. Gallagher, Z. Zhang, and Z. Tu, "Deeply-supervised nets," in *AISTATS*, 2015.

[93] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *ICML*, 2015.

[94] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *ICML*, 2010.

[95] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *AISTATS*, 2010.

[96] L. Yu, X. Yang, H. Chen, J. Qin, and P.-A. Heng, "Volumetric convnets with mixed residual connections for automated prostate segmentation from 3D MR images," in *AAAI*, 2017.

[97] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *ACM MM*, 2014.

[98] P. Gravel, G. Beaudoin, and J. A. De Guise, "A method for modeling noise in medical images," *TMI*, vol. 23, no. 10, pp. 1221–1232, 2004.

[99] A. A. Lasboo, P. Rezai, and V. Yaghmai, "Morphological analysis of pancreatic cystic masses," *Academic radiology*, vol. 17, no. 3, pp. 348–351, 2010.

[100] L. Zhang, L. Lu, R. M. Summers, E. Kebebew, and J. Yao, "Personalized pancreatic tumor growth prediction via group learning," in *MICCAI*, 2017.

[101] Q. Dou, H. Chen, L. Yu, J. Qin, and P.-A. Heng, "Multilevel contextual 3-d cnns for false positive reduction in pulmonary nodule detection," *IEEE Transactions on Biomedical Engineering*, vol. 64, no. 7, pp. 1558–1567, 2016.

[102] S. Hussein, M. M. Chuquicusma, P. Kandel, C. W. Bolan, M. B. Wallace, and U. Bagci, "Supervised and unsupervised tumor characterization in the deep learning era," *arXiv:1801.03230*, 2018.

[103] A. C. Society, *Cancer facts & figures 2019*, 2019.

[104] Y. Wu and K. He, "Group normalization," in *ECCV*, 2018.

[105] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *CVPR*, 2017.

[106] Q. Yu, L. Xie, Y. Wang, Y. Zhou, E. K. Fishman, and A. L. Yuille, "Recurrent saliency transformation network: Incorporating multi-stage visual cues for small organ segmentation," in *CVPR*, 2018.

[107] J. C. Scatarige, E. K. Fishman, F. P. Kuhajda, G. A. Taylor, and S. S. Siegelman, "Low attenuation nodal metastases in testicular carcinoma.," *Journal of computer assisted tomography*, vol. 7, no. 4, pp. 682–687, 1983.

[108] T. Leong, C. Everitt, K. Yuen, S. Condron, A. Hui, S. Y. Ngan, A. Pitman, E. W. Lau, M. MacManus, D. Binns, *et al.*, "A prospective study to evaluate the impact of fdg-pet on ct-based radiotherapy treatment planning for oesophageal cancer," *Radiotherapy and oncology*, vol. 78, no. 3, pp. 254–261, 2006.

[109] R. Girshick, "Fast r-cnn," in *ICCV*, 2015.

[110] K. Yan, Y. Peng, V. Sandfort, M. Bagheri, Z. Lu, and R. M. Summers, "Holistic and comprehensive annotation of clinically significant findings on diverse ct images: Learning from radiology reports and label ontology," in *CVPR*, 2019.

[111] H. J. Kuijf, J. M. Biesbroek, J. De Bresser, R. Heinen, S. Andermatt, M. Bento, M. Berseth, M. Belyaev, M. J. Cardoso, A. Casamitjana, *et al.*, "Standardized assessment of automatic segmentation of white matter hyperintensities and results of the wmh segmentation challenge," *TMI*, vol. 38, no. 11, pp. 2556–2568, 2019.

[112] B. H. Menze, A. Jakab, S. Bauer, J. Kalpathy-Cramer, K. Farahani, J. Kirby, Y. Burren, N. Porz, J. Slotboom, R. Wiest, *et al.*, "The multimodal brain tumor image segmentation benchmark (brats)," *TMI*, vol. 34, no. 10, pp. 1993–2024, 2014.

[113] D. Jin, D. Guo, T.-Y. Ho, A. P. Harrison, J. Xiao, C.-K. Tseng, and L. Lu, "Accurate esophageal gross tumor volume segmentation in pet/ct using two-stream chained 3d deep network fusion," in *MICCAI*, 2019.

[114] L. Xu, G. Tetteh, J. Lipkova, Y. Zhao, H. Li, P. Christ, M. Piraud, A. Buck, K. Shi, and B. H. Menze, "Automated whole-body bone lesion detection for multiple myeloma on 68ga-pentixafor pet/ct imaging using deep learning methods," *Contrast media & molecular imaging*, vol. 2018, 2018.

[115] X. Zhao, L. Li, W. Lu, and S. Tan, "Tumor co-segmentation in pet/ct using multi-modality fully convolutional neural network," *Physics in Medicine & Biology*, vol. 64, no. 1, p. 015 011, 2018.

[116] C. R. Maurer, R. Qi, and V. Raghavan, "A linear time algorithm for computing exact euclidean distance transforms of binary images in arbitrary dimensions," *TPAMI*, vol. 25, no. 2, pp. 265–270, 2003.

[117] L. Liu, H. Jiang, P. He, W. Chen, X. Liu, J. Gao, and J. Han, "On the variance of the adaptive learning rate and beyond," *arXiv:1908.03265*, 2019.

[118] A. P. Harrison, Z. Xu, K. George, L. Lu, R. M. Summers, and D. J. Mollura, "Progressive and multi-path holistically nested neural networks for pathological lung segmentation from ct images," in *MICCAI*, 2017.

[119] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv:1412.6980*, 2014.

[120] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *CVPR*, 2018.

[121] K. Sun, B. Xiao, D. Liu, and J. Wang, "Deep high-resolution representation learning for human pose estimation," in *CVPR*, 2019.

[122] Z. Zhu, Y. Lu, W. Shen, E. K. Fishman, and A. L. Yuille, "Segmentation for classification of screening pancreatic neuroendocrine tumors," *arXiv:2004.02021*, 2020.

[123] Z. Zhu, K. Yan, D. Jin, J. Cai, T.-Y. Ho, A. P. Harrison, D. Guo, C.-H. Chao, X. Ye, J. Xiao, *et al.*, "Detecting scatteredly-distributed, small, and critically important objects in 3d oncology imaging via decision stratification," *arXiv:2005.13705*, 2020.

[124] S. Fu, Y. Lu, Y. Wang, Y. Zhou, W. Shen, E. Fishman, and A. Yuille, "Domain adaptive relational reasoning for 3d multi-organ segmentation," in *MICCAI*, 2020.

Vita

Zhuotun Zhu is completing his Ph.D. degree of Computer Science at the Johns Hopkins University, advised by Bloomberg Distinguished Professor Alan L. Yuille. He has been interested in computer vision and machine learning, especially dedicated to the medical image analysis during his Ph.D. research. Before that, he obtained Master of Science degree in Statistics from University of California, Los Angeles in 2016, and Bachelor of Engineering degree in Electronics and Information Engineering from Huazhong University of Science and Technology in 2015. He was introduced to computer vision and machine learning when he was a sophomore and has been gradually fascinated by the magical research world since then. As a Ph.D. student, he was devoted to the volumetric medical images segmentation, especially targeted on the early detection of pancreatic cancers, founded by the Lustgarten Foundation. He interned in PAII, NVIDIA, and Microsoft Research, where he was very fortunate to work with so many amazing and professional people from the industry. He was a recipient of the MICCAI 2020 NIH award.